

UNIVERSIDADE FEDERAL DO PARANÁ

DOUGLAS NATAN MEIRELES CARDOSO

FELLIPE ALEXANDRE

SBIM: SISTEMA GERENCIADOR DE IMAGENS BIOLÓGICAS NA WEB

CURITIBA

2013

DOUGLAS NATAN MEIRELES CARDOSO

FELLIPE ALEXANDRE

SBIM: SISTEMA GERENCIADOR DE IMAGENS BIOLÓGICAS NA WEB

Trabalho de conclusão de curso apresentado como requisito parcial à conclusão do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas. Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná.

Orientador: Prof. Dr. Luiz Antônio Pereira Neves

CURITIBA

2013

TERMO DE APROVAÇÃO

DOUGLAS NATAN MEIRELES CARDOSO

FELLIPE ALEXANDRE

Trabalho de Conclusão de Curso aprovado como requisito para aprovação na disciplina "Trabalho de Conclusão de Curso" no curso de Tecnologia em Análise e Desenvolvimento de Sistemas, Setor de Educação Profissional e Tecnológica, Universidade Federal do Paraná, pela seguinte banca examinadora:

Prof. Dr. Luiz Antônio Pereira Neves
Orientador

Profa. M^a. Rafaela M. Fontana
Membro

Prof. Dr. Roberto Tadeu Raittz
Membro

M^a Terumi Paula Bonfim Kamada
Membro

Curitiba, 12 de março de 2013.

AGRADECIMENTOS

Primeiramente, agradecemos a Deus por permitir que concluíssemos mais essa etapa em nossas vidas em meio a tantas dificuldades.

Aos nossos pais e amigos que direta e indiretamente nos ajudou nos dando apoio e estando ao nosso lado.

Ao orientador Prof. Dr. Luiz Antônio Pereira Neves por sempre nos incentivar a buscar melhores resultados e pela dedicação em sempre atender todos os seus alunos sem qualquer distinção.

RESUMO

Os sistemas para gerenciamento e compartilhamento de imagens biológicas hoje disponíveis, oferecem basicamente uma plataforma para armazenamento e troca de dados e informações via Web. Uma análise da literatura nos mostra que sistemas com este perfil, em geral, possui poucos recursos para processamento e análise de imagens. Assim, com o objetivo de oferecer uma alternativa aos sistemas disponíveis na Web que integre compartilhamento e manipulação de imagens biológicas, foi desenvolvido o *Shared Biological Image Manager* (SBIM). Antes de sua construção, foi feito um estudo para identificar quais serviços o sistema deveria oferecer, bem como para ajudar na escolha de uma plataforma de desenvolvimento capaz de atender as necessidades da aplicação. A linguagem de programação escolhida foi JAVA juntamente com seus *frameworks* adjacentes, utilizando *Postgresql* como sistema gerenciador de bancos de dados e padrão de desenvolvimento *MVC (Model View Controller)*. Além de oferecer recursos que se destacam em relação aos sistemas existentes, uma das principais funcionalidades do SBIM consiste em permitir aos seus usuários classificar e recuperar imagens, utilizando algoritmos de aprendizagem através do uso do *Support Vector Machine* (SVM). Os testes do sistema proposto foram baseados em metodologias específicas para sua avaliação, utilizando Pressman.

ABSTRACT

The available systems to sharing and management of medical images are basically a platform to storage, data and information changes through the Web. A review in the literature shows to us that the processing resources and images analysis are very limited. Then, with the goal to offer a better tool then previous systems that can join sharing and manipulation of medical images, we have done the Shared Biological Image Manager (SBIM). Before the building, we have made a study to identify which services the application could have and also to better choose one platform capable to correspond with our needs. The computer language chosen was JAVA joined with its adjacent frameworks, using Postgresql as data base manager and finally MVC (Model View Controller) as development pattern. The SBIM's highlight is to make possible for its users, to classify and to retrieve images from a data base based on artificial intelligence using Support Vector Machine. The system's tests were based on specifics methodologies to its evaluation using Pressman.

SUMÁRIO

1	INTRODUÇÃO.....	10
1.1	OBJETIVO GERAL	10
1.2	OBJETIVOS ESPECÍFICOS	10
1.3	PROBLEMATIZAÇÃO	11
1.4	JUSTIFICATIVA	11
1.5	ESTRUTURA DO TRABALHO	12
2	REVISÃO DE LITERATURA.....	13
2.1	CBIR (Content Based Image Retrieval).....	13
2.2	IMAGEM.....	13
2.3	ALGORITIMOS DE APRENDIZAGEM	18
2.4	REDES NEURAIS	19
2.5	SUPPORT VECTOR MACHINE	21
3	METODOLOGIA	26
3.1	MODELO DE PROCESSO DE ENGENHARIA DE SOFTWARE	26
3.1.1	PLANO DE ATIVIDADE	26
3.1.2	PLANO DE RISCOS	29
3.2	MATERIAIS	29
3.2.1	AMBIENTE DE SOFTWARE	29
3.2.2	AMBIENTE DE HARDWARE.....	33
3.3	DESENVOLVIMENTO DO PROJETO	34
3.3.1	ETAPA 1 - MODELAGEM DO SISTEMA.....	34
3.3.2	ETAPA 2 - MODELAGEM DO BANCO DE DADOS.....	39
3.3.3	ETAPA 3 - ADAPTAÇÕES DAS FERRAMENTAS	39
3.3.4	ETAPA 4- USO DA INTELIGÊNCIA ARTIFICIAL	40
3.3.5	ETAPA 5 - IMPLEMENTAÇÃO DA MODELAGEM PROPOSTA	42
3.3.6	ETAPA 6 - PROTOCOLO DE VALIDAÇÃO.....	45
4	APRESENTAÇÃO DO SOFTWARE.....	49
4.1	TELAS DO SISTEMA	49
4.2	MANUAL DE INSTALAÇÃO	57
4.2.1	PASSO 1 – CONFIGURAR O BANCO DE DADOS	57
4.2.2	PASSO 2 – INSERIR O CÓDIGO FONTE NO APACHE TOMCAT	59
4.2.3	PASSO 3 – ACESSAR A PÁGINA INICIAL DO SISTEMA SBIM	59

5 CONCLUSÃO	60
REFERÊNCIAS	62
APÊNCIDE A - DIAGRAMA DE CASO DE USO	64
APÊNCIDE B - DIAGRAMA DE CLASSE	71
APÊNCIDE C - DIAGRAMA ENTIDADE RELACIONAMENTO	72
APÊNCIDE D - DIAGRAMA DE SEQUENCIA	73
APÊNCIDE E - DICIONÁRIO DE DADOS	85
APÊNCIDE F - DIAGRAMA DE CLASSES DE IMPLEMENTAÇÃO	91
APÊNCIDE G - TABELAS DE ANÁLISE DE RISCOS	102
APÊNCIDE H - CASOS DE TESTE	105
APÊNCIDE I - DIAGRAMA DE ATIVIDADES	116

LISTA DE FIGURAS

Figura 1 -	Ilustração da Tela do Computador.	14
Figura 2 -	Erro de reconstrução versus tamanho das subimagens.....	16
Figura 3 -	Exemplo de imagem transformada.....	17
Figura 4 -	Componentes de um sistema de IA.....	18
Figura 5 -	Modelo matemático para um neurônio	19
Figura 6 -	Exemplo de uma rede neural	20
Figura 7 -	Treinamento bidimensional	22
Figura 8 -	Treinamento tridimensional	23
Figura 9 -	Treinamento Linear	23
Figura 10 -	Calculo da distância entre os hiperplanos	25
Figura 11 -	WBS do SBIM.....	27
Figura 12 -	Diagrama de Gantt	28
Figura 13 -	Exemplo de imagens da base face.....	46
Figura 14 -	Exemplo de imagens da base íris.....	47
Figura 15 -	Exemplo de imagens da base flores.....	47
Figura 16 -	Exemplo de imagens da base palma da mão.....	48
Figura 17 -	Registro passo 1.....	49
Figura 18 -	Registro passo 2.....	50
Figura 19 -	Autenticação do usuário	50
Figura 20 -	Tela principal	51
Figura 21 -	Envio de mensagens	51
Figura 22 -	Perfil do pesquisador.....	52
Figura 23 -	Listagem de convites.....	52
Figura 24 -	Listagem de contatos	53
Figura 25 -	Listagem de bases	53
Figura 26 -	Gerenciamento de base	54
Figura 27 -	Edição de imagem.....	55
Figura 28 -	Classificação de imagem.....	56
Figura 29 -	Busca de imagem.....	56
Figura 30 -	Nova database	57
Figura 31 -	Restauração de backup.....	58
Figura 32 -	Estrutura de tabelas	58

Figura 33 -	Estrutura de pastas	59
-------------	---------------------------	----

LISTA DE TABELAS

Tabela 1 -	Tabela de divisão de tarefas	43
Tabela 2 -	Tabela de acurácia do SVM	46
Tabela 3 -	Tempos de classificação e treinamento	48

1 INTRODUÇÃO

Uma das principais vantagens que a Web trouxe foi a possibilidade de interagir com pessoas de vários lugares do mundo, compartilhar documentos, imagens e vídeos, deixando-os disponíveis em qualquer lugar, desde que haja conexão com a internet.

Com o desenvolvimento das ferramentas para criação de conteúdo da internet, a forma de interação com o usuário também foi se desenvolvendo, assim as páginas da Web deixaram de ter conteúdo estático passando para outro nível de interatividade. A partir deste ponto o desenvolvedor não é mais o único criador de conteúdo, agora o usuário também tem este poder.

Embora o advento da internet tenha evoluído de uma maneira exponencial, pouco se houve falar em gerenciadores de imagem na nuvem. Gerenciadores de imagens são sistemas que facilitam o armazenamento, edição, organização e compartilhamento de imagens. No Brasil existem muitos projetos acadêmicos espalhados por toda a parte ligados ao estudo de processamento de imagens digitais. Uma forma de aproximar os diversos integrantes destes projetos é o tema proposto nesse trabalho.

O foco deste projeto é criar um gerenciador de imagens médicas e biológicas na Web capaz de permitir total integração entre seus usuários, compartilhamento e manipulação de imagens, utilizando conceitos de inteligência artificial.

1.1 OBJETIVO GERAL

Desenvolver uma ferramenta que facilite a edição e o compartilhamento de imagens biológicas entre pesquisadores de diferentes instituições, utilizando a plataforma Web.

1.2 OBJETIVOS ESPECÍFICOS

- Desenvolver uma rede social, permitindo a troca de mensagens entre usuários, criação de bases de imagens, convites para contato e participação de bases;

- Adequar a interface do sistema, baseando-se nos conceitos de ergonomia para Web.
- Desenvolver ferramentas para marcação e edição de imagens;
- Desenvolver um algoritmo de classificação e de recuperação de imagens.

1.3 PROBLEMATIZAÇÃO

Atualmente as trocas de imagens entre pesquisadores de diversas universidades e grupos de pesquisa ocorrem através de e-mails. Esta é uma forma de trabalho rudimentar se pensarmos na quantidade de recursos computacionais existente nos dias atuais. Em posse disso, foi proposto um sistema que pudesse melhorar a metodologia de trabalho dos pesquisadores chamado de SBIM.

A primeira versão deste sistema, apresentado por Neves, Porcides, Giraldi, no ano de 2011, apresentava alguns problemas de interface, instabilidade do sistema, difícil navegabilidade e principalmente a arquitetura computacional escolhida não era adequada para os padrões do sistema, o que inviabilizava as manutenções futuras.

Portanto, as problematizações encontradas nesta pesquisa baseando-se no sistema SBIM anterior são:

- Problemas de instabilidade e segurança;
- Modelagem computacional inadequada;
- Interface não atrativa ao usuário;
- Poucos recursos para a edição de imagens;
- Falta de padrão de desenvolvimento;

1.4 JUSTIFICATIVA

A falta de ferramentas que unam edição de imagens e interação entre pesquisadores via Web foram as principais razões que motivaram o desenvolvimento deste projeto. Antes, o estudo de uma mesma imagem entre vários pesquisadores era feito através de troca de e-mails entre os envolvidos como dito na

seção anterior. Nesta forma de trabalho, a produtividade no tratamento das informações e até mesmo no estudo das imagens é bem pequena.

A plataforma Web foi escolhida graças à possibilidade de compartilhamento de dados com pessoas de qualquer parte do mundo desde que haja conexão com a internet. Mediante isso, é possível conectar diversos pesquisadores de diferentes instituições, interessados em compartilhar informações sobre um tema em comum. A escolha desta plataforma também foi influenciada pelo fato de permitir o acesso via dispositivos móveis, independente do sistema operacional utilizado.

1.5 ESTRUTURA DO TRABALHO

A organização deste trabalho está dividida sob os seguintes capítulos:

Capítulo 2: Revisão de literatura. Este capítulo contém alguns conceitos de processamento de imagens e algoritmos de aprendizagem, fundamentalmente sobre Support Vector Machine. Estes são a base da fundamentação teórica deste trabalho.

Capítulo 3: Metodologia. Este capítulo contém as etapas da construção do projeto, o funcionamento do sistema, como ocorreu o desenvolvimento do tema proposto utilizando os conceitos apresentados na fundamentação teórica e a validação do sistema através do protocolo de validação.

Capítulo 4: Apresentação do Software. Este capítulo apresenta o funcionamento do sistema, mostrando as telas, o manual de instalação e por fim as considerações finais deste trabalho.

Capítulo 5: Conclusão. Este capítulo nos mostra os resultados obtidos mediante o que foi proposto nos objetivos do trabalho, e quais serão os estudos futuros.

Apêndice: Contém os diagramas do sistema.

2 REVISÃO DE LITERATURA

Neste capítulo apresentamos o estudo bibliográfico deste trabalho, definindo a fundamentação teórica que é a base para o entendimento de como foi implementada a classificação e recuperação de imagens dentro do sistema, além dos artigos científicos analisados.

2.1 CBIR (CONTENT BASED IMAGE RETRIEVAL)

Com a evolução da tecnologia digital, há um aumento significativo e constante de imagens armazenadas em formato eletrônico. Tal armazenagem facilita a busca de imagens para pesquisa, reconhecimento de pessoas, padrões etc. Atualmente, é possível buscar imagens através de diversos sites populares de pesquisa, como o Google, por exemplo. Ao fazer uma pesquisa, existem diferentes maneiras de buscar imagens: através do nome, por metadados, por palavras chaves, ou por meio de suas características. A esta última damos o nome de *Content Based Image Retrieval* (CBIR), que no português significa Recuperação de Imagens Baseado em Conteúdo.

Toshikazu Kato foi o primeiro pesquisador a trazer a ideia de CBIR, em 1992, através de experimentos que objetivavam recuperar imagens a partir do banco de dados levando em conta as cores e principais características. A partir daí um número crescente de sistemas com essas mesmas características têm surgido. Em geral, para que um sistema seja classificado como CBIR, a extração de características das imagens deve ser predominantemente automática, ou seja, não ter intervenção do usuário (Eakins, Graham, 2002, p.23).

2.2 IMAGEM

Uma imagem nada mais é do que uma função bidimensional de intensidade da luz $f(x,y)$ onde x e y determinam coordenadas espaciais e o valor f , em qualquer ponto da imagem, se refere ao nível de cinza (brilho) daquele ponto (GONZALEZ; WOODS, 2000, p.21). As imagens que nós percebemos em atividades visuais do dia-a-dia, consistem na luz refletida dos objetos. A natureza básica de $f(x,y)$ pode ser caracterizada por dois componentes: (1) a quantidade de luz incidindo na cena que está sendo observada e (2) a quantidade de luz refletida pelos objetos na cena. Esses componentes são chamados de iluminação e reflectância,

respectivamente e matematicamente são representados por $i(x, y)$ e $r(x, y)$ (*ibidem*). O produto dessas funções resulta em $f(x, y)$.

De modo geral, os valores dos pontos de uma imagem obtidos a partir da função $f(x, y)$ são chamados de elementos da figura, em inglês *Picture Elements*, ou comumente conhecido como pixel. Cada pixel é representado em um vetor bidimensional (matriz), e cada posição do vetor assume um valor entre 0 e 255. Existe ainda uma diferença de representação para imagens em tom de escala de cinzas (um canal) ou para imagens coloridas com três canais (RGB). Para o primeiro caso o pixel assume um valor entre 0 e 255, e para o segundo caso três valores entre 0 e 255 que representam exatamente os valores de *Red*, *Green* e *Blue* respectivamente.

Para ilustração, imaginemos a tela de um computador de resolução $X \times Y$ como mostra a figura 1. A matriz que representa a imagem mostrada na tela tem Y linhas e X colunas. Cada pixel possui quatro vizinhos, dois horizontais e dois verticais. À medida que o número de pixels aumenta, a intensidade de luz na imagem fica maior o que aumenta a sua qualidade.

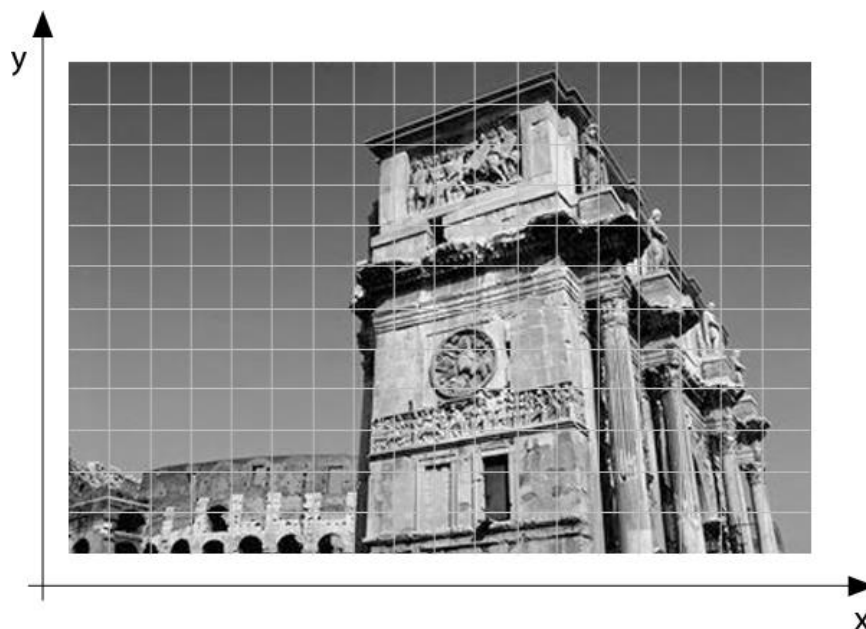


Figura 1 - Ilustração da Tela do Computador.

2.2.1 COMPRESSÃO DE IMAGENS

Em muitos casos, armazenar uma imagem digital é inviável devido a grande quantidade de dados que é produzido para sua criação. Segundo Gonzales, é

preciso mais de 25 Gigabytes de dados para representar a *Encyclopaedia Britannica* na forma digital por exemplo.

Uma forma de representar uma imagem em uma dimensão menor conhecida como compressão de imagens que tem como efeito geral a diminuição das redundâncias dos dados de uma imagem, tornando possível transmitir os mesmos dados da imagem original de maneira eficiente. Do ponto de vista matemático, isto corresponde a transformar uma matriz de pixels de duas dimensões num conjunto de dados estatisticamente descorrelacionado (Gonzales, Woods, 2010, p.218).

Podemos destacar dois tipos de compressão: compressão livre de erros e compressão com perdas. Para o primeiro tipo, a aplicabilidade é observada em aplicações que armazenam dados minuciosos e não tolera perda de critérios que comprometa de alguma forma o resultado esperado, um exemplo disso são sistemas de arquivamento de documentos médicos. Em contrapartida existem aplicações onde há uma tolerância a perdas e a preocupação maior está no tempo de processamento e na economia de espaço no armazenamento.

Existem várias técnicas de compressão de imagem, mas compressão de imagens por transformada é o que tem sido extensivamente estudado principalmente “sistemas de codificação por transformada” baseado nas transformadas de Karhunen-Loève (KLT), Fourier (FFT), Discreta do Cosseno (DCT) e Walsh-Hadamard (WVT) (Gonzales, Woods, 2010, p.267).

Em geral, o que difere cada uma delas é a capacidade de gerar coeficientes descorrelacionados, concentrando a maior parte da energia dos pontos de intensidade de luz da imagem em um número reduzido de coeficientes. Para a escolha da transformada, deve-se levar em conta qual a quantidade de erro de reconstrução que pode ser tolerado no sistema. Ao aplicar uma transformada em uma imagem, uma parte de seus coeficientes são desconsiderados, restando apenas os coeficientes significativos. Ao descartar uma parte dos coeficientes devemos considerar que quanto maior for o descarte maior serão os erros na recuperação da imagem a partir de suas subimagens.

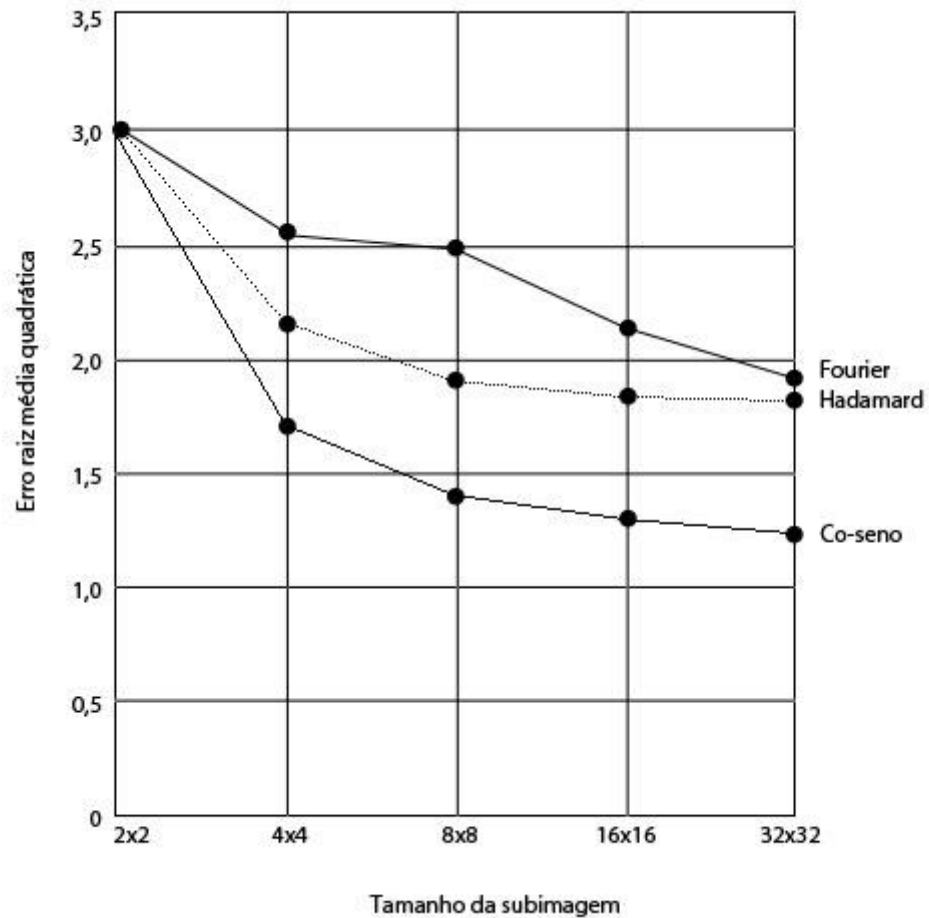


Figura 2 - Erro de reconstrução versus tamanho das subimagens (Gonzales, Woods, 2010, p.271).

2.2.2 COMPRESSÃO DE IMAGENS COM TRANSFORMADA DISCRETA DO COSSENO

A transformada discreta cosseno (DCT) é uma função linear e inversível, $\mathbf{R} \rightarrow \mathbf{R}$, que expressa sinais como uma soma de funções cosseno discretas. O sinal original é convertido para o domínio da frequência pela DCT, e é possível converter o sinal de volta para o domínio do tempo aplicando-se a DCT inversa ou IDCT (Matos, 2008, p.24). Em 1 e 2, apresentamos a fórmula da DCT bidimensional e em 3 e 4 apresentamos a fórmula da IDCT.

$$F(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{a-1} \sum_{y=0}^{b-1} f(x, y) \cos\left(\frac{(2x+1)u\pi}{2N}\right) \cos\left(\frac{(2y+1)v\pi}{2N}\right)$$

(1)

$$\alpha(u), \alpha(v) = \begin{cases} \sqrt{\frac{1}{N}}, & \text{para } u, v = 1 \\ \sqrt{\frac{2}{N}}, & \text{para os demais} \end{cases}, \quad (2)$$

$$P(u, v) = \frac{1}{4} \sum_{x=0}^{a-1} \sum_{y=0}^{b-1} \alpha(x) \alpha(y) F(x, y) \cos\left(\frac{(2u+1)x\pi}{2N}\right) \cos\left(\frac{(2v+1)y\pi}{2N}\right) \quad (3)$$

$$\alpha(x) \alpha(y) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{para } x, y = 0 \\ 1, & \text{para } x, y > 0 \end{cases} \quad (4)$$

A figura 3 ilustra o impacto do tamanho das subimagens no erro de reconstrução da codificação por transformada.



Figura 3 - (a) imagem original, (b) imagem recuperada utilizando 50% dos coeficientes DCT, (c) imagem recuperada utilizando 25% dos coeficientes DCT e (d) imagem recuperada utilizando 12,5% dos coeficientes DCT.

2.3 ALGORITMOS DE APRENDIZAGEM

Um dos objetivos da Inteligência Artificial é o desenvolvimento de sistemas computacionais que representem o modelo de funcionamento, e que manifestem o comportamento intelectual do ser humano na realização de uma determinada atividade (CONDUTA, MAGRIN, 2010, p.2). Algoritmos de aprendizagem vêm de encontro com essa definição.

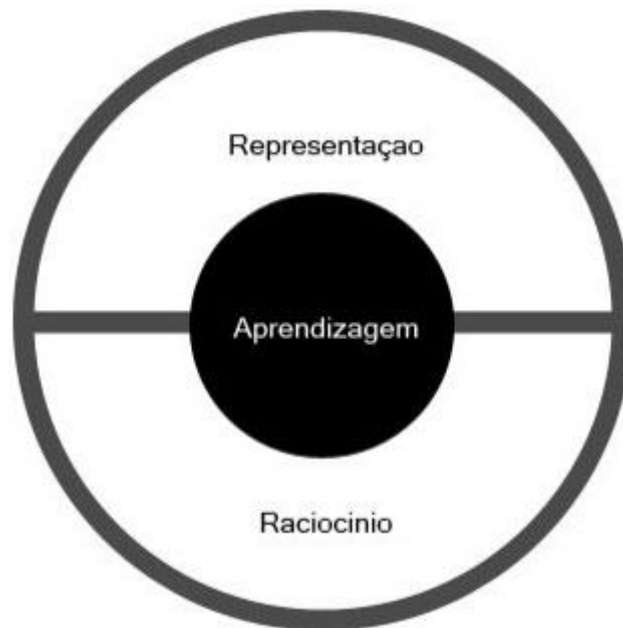


Figura 4 - Componentes de um sistema de IA – Adaptado de Haykin (2002).

O campo de aprendizagem de máquina costuma distinguir três casos: aprendizagem supervisionada, aprendizagem não supervisionada e aprendizagem por reforço. Nesse trabalho, daremos foco na aprendizagem supervisionada.

Para exemplificação, propomos a seguinte situação: Você está disposto a ensinar uma criança, ainda na fase das descobertas, o que é um ônibus e a estratégia adotada para isso é fazê-la reconhecer através de fotos. Quanto mais fotos de ônibus você mostrar para ela e dizer que imagens que possuem aquelas características representam um ônibus, tão logo a criança saberá reconhecer com facilidade o que é um ônibus. Para este tipo de aprendizagem damos o nome de supervisionada. O problema desta envolve a aprendizagem de uma função a partir de exemplos de suas entradas e saídas (RUSSEL, NORVING. 2004, p. 630). Se pensarmos na ilustração anterior, a primeira condição para que a criança pudesse aprender foi à necessidade de ter exemplos do mundo real para abstração do

mesmo. É preciso ter um cuidado muito grande nesse tipo de aprendizagem na hora de fornecer um conjunto de treinamento, pois é a partir dele que será realizada a comparação do mundo real com o que foi aprendido e em alguns sistemas esse tipo de aplicação não é indicado.

Na próxima seção falaremos sobre dois algoritmos de aprendizagem supervisionada: Redes Neurais Artificiais e Support Vector Machine.

2.4 REDES NEURAIS

Um neurônio é uma célula no cérebro cuja principal função é coletar, processar e disseminar sinais elétricos. Acredita-se que a capacidade de processamento de informações do cérebro emerge principalmente de redes de neurônios. Por essa razão, uma parte do trabalho inicial de Inteligência Artificial teve como objetivo criar redes neurais artificiais para a aprendizagem de máquinas (RUSSEL; NORVING, 2004, p. 713). Na figura abaixo temos um modelo matemático simples do neurônio, criado por McCulloch e Pitts (1943).

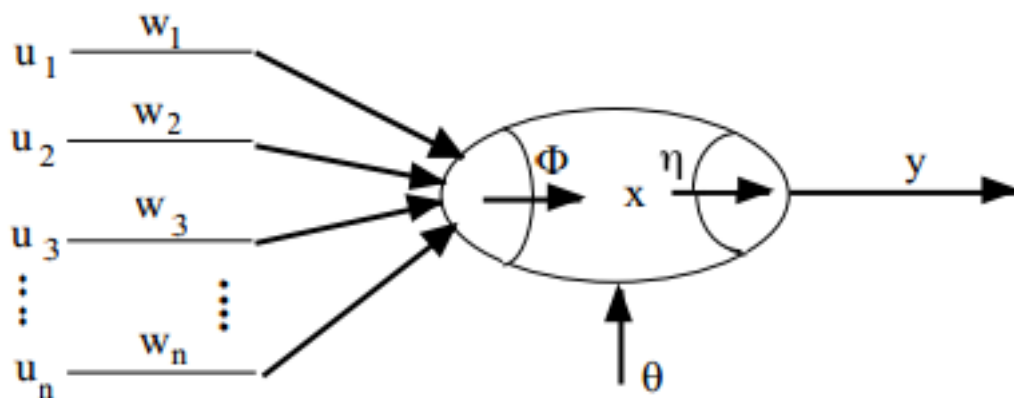


Figura 5 - Modelo matemático para um neurônio. (NORVING, RUSSEL, 2004, p.714).

Neste modelo as entradas $W_i U_i$ são combinadas usando uma função Φ para produzir um estado de ativação do neurônio que, através da função η , vai produzir a saída do neurônio (correspondente à frequência de descarga do neurônio biológico). Um valor auxiliar θ é geralmente usado para representar uma polarização, valor abaixo do qual a saída é nula (BARRETO, JORGE, 2002, p. 14).

Existem duas categorias principais de estruturas de redes neurais: redes de alimentação direta e redes recorrentes. Uma rede de alimentação direta representa uma função de sua entrada atual; desse modo, ela não tem nenhum estado interno além dos pesos propriamente ditos. Neurônios que têm sua saída como saída da rede, pertencem à camada de saída, ou última camada. Neurônios que não pertencem à camada de entrada, nem à de saída, são neurônios internos à rede, podendo se organizar em uma ou mais camadas internas (BARRETO, JORGE, 2002, p. 25). Na figura seguinte, temos uma rede neural simples com duas unidades de entrada, duas unidades ocultas e uma unidade de saída.

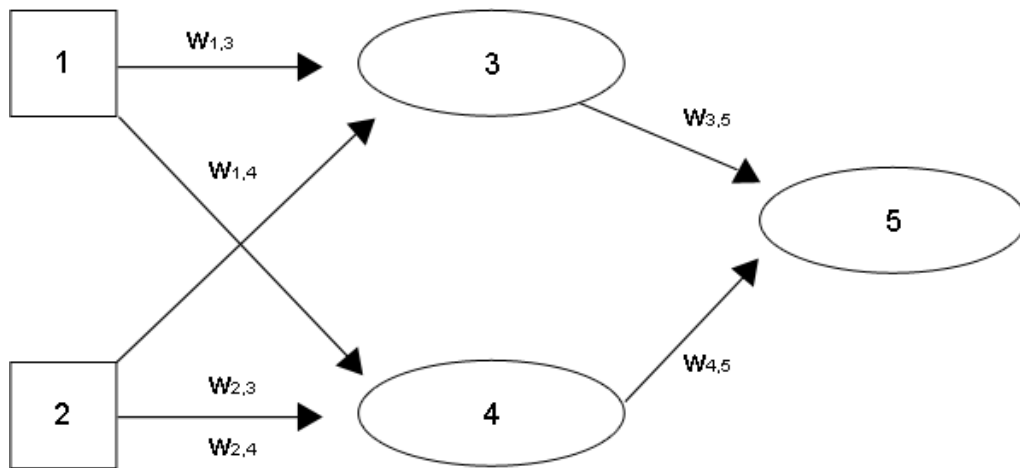


Figura 6 - Exemplo de uma rede neural muito simples com duas entradas, uma camada oculta de duas unidades e uma saída.

Para analisar mais de perto a asserção de que uma rede de alimentação direta representa uma função de suas entradas, faremos uso da explicação extraída do livro Inteligência Artificial, página 715, por Russell e Norving (Russel, Norving, 2004). Considere a rede simples mostrada na figura anterior e imaginemos um vetor de entrada $x = (x_1, x_2)$, as ativações das unidades de entrada são definidas como $(a_1, a_2) = (x_1, x_2)$ e a rede calcula:

$$a_5 = g(w_{3,5}a_3 + w_{4,5}a_4) = g\left(w_{3,5}g(w_{1,3}a_1 + w_{2,3}a_2) + w_{4,5}g(w_{1,4}a_1 + w_{2,4}a_2)\right). \quad (5)$$

Isto é, expressando a saída de cada unidade oculta como uma função de suas entradas, mostramos que a saída da rede como um todo, a_5 , é uma função das

entradas da rede. Além disso, vemos que os pesos na rede atuam como parâmetros dessa função; escrevendo-se W para os parâmetros, a rede calcula uma função $Hw(x)$. Ajustando os pesos, mudamos a função que a rede representa. É assim que a aprendizagem ocorre em redes neurais. Uma rede pode ser utilizada para a classificação e recuperação de dados. (RUSSEL; NORVING, 2004, p. 715).

2.5 SUPPORT VECTOR MACHINE

Até aqui a discussão sobre algoritmos de aprendizagem se resumiu em redes neurais. Nesta seção temos outra classe de redes conhecida como máquinas de vetor de suporte ou do inglês *Support Vector Machine*.

O SVM é embasado pela teoria de aprendizagem estatística desenvolvida por V. N. Vapnik em 1995 (LORENA; CARVALHO, 2007, p.1). Essa técnica tem recebido bastante atenção principalmente por sua utilização ser mais fácil às redes neurais artificiais. O treinamento tanto para a solução de problemas lineares ou não lineares é feito da mesma forma.

No SVM, a tarefa de classificação normalmente envolve a separação de dados em conjuntos de treinamento e testes. Cada instância no conjunto de treinamento contém um valor de destino (rótulos da classe) e seus respectivos atributos. O objetivo do SVM é de produzir um modelo baseado no conjunto de treinamento que prediga o valor de destino com os conjuntos de testes dando apenas os atributos do conjunto de testes.

Temos que o SVM requer a solução do problema a seguir (HSU; CHANG; LIN, 2010, p.1):

$$\begin{aligned} \min_{w,b,\varepsilon} \quad & \frac{1}{2} w^T w + C \sum_{i=1}^m \varepsilon_i \\ \text{sujeito a} \quad & y_1(w^T \Phi(x_1) + b) \geq 1 - \varepsilon_i, \\ & \varepsilon_i \geq 0. \end{aligned} \tag{6}$$

Dado um conjunto de treinamento S em que:

$$S = \{(y_1, x_1), \dots, (y_m, x_m)\}, x \in \mathbb{R}^n, y \in \{-1, 1\} \quad (7)$$

Temos o treinamento do conjunto S levando em consideração os rótulos das classes y_m e o respectivo valor de seu atributo x_m . Os vetores de treinamento são mapeados em um grande espaço de dimensão pela função Φ . O SVM busca uma separação linear no hiperplano com a máxima margem nesse grande espaço de dimensão (HSU; CHANG; LIN, 2010, p.2).

Dizemos que um classificador é linear somente quando ele é separado por um hiperplano. Considerando novamente o conjunto S e seus respectivos rótulos, S é linearmente separável se for possível separar os dados das classes 1 e -1 por um hiperplano (SCHÖLKOPF; SMOLA, 2001, p.14).

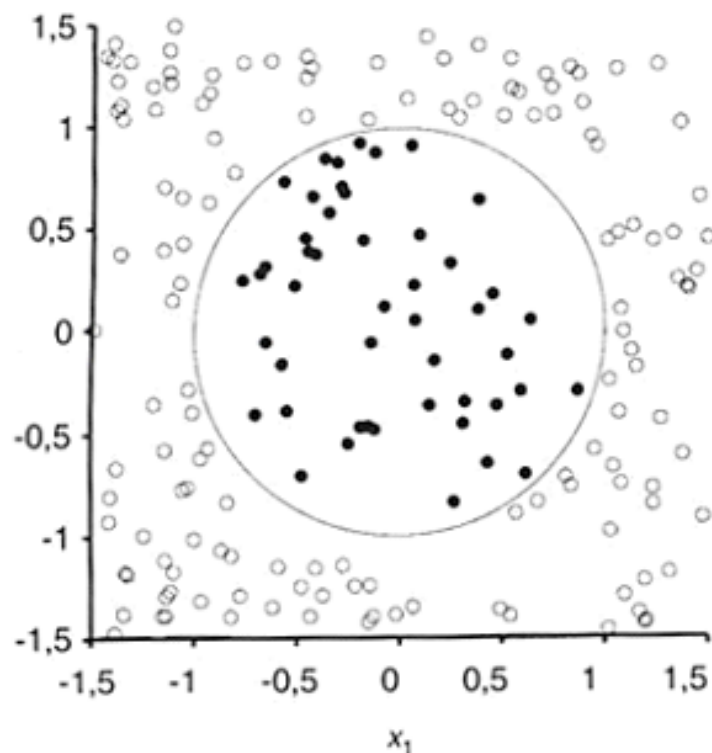


Figura 7 - Um treinamento bidimensional com exemplos positivos representados como círculos pretos e exemplos negativos como círculos brancos. O verdadeiro limite de decisão, $x_1^2 + x_2^2 \leq 1$, também é mostrado (NORVING, RUSSEL, 2004, p.714).

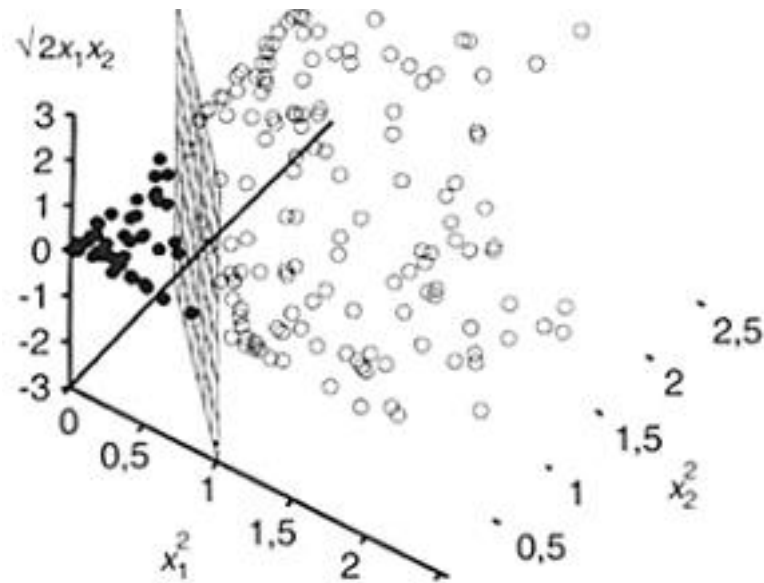


Figura 8 - Um treinamento tridimensional $(x_1^2, x_2^2, \sqrt{2}x_1x_2)$ com exemplos positivos representados como círculos pretos e exemplos negativos como círculos brancos. O verdadeiro limite de decisão, $x_1^2 + x_2^2 \leq 1$, também é mostrado (NORVING, RUSSEL, 2004, p.714).

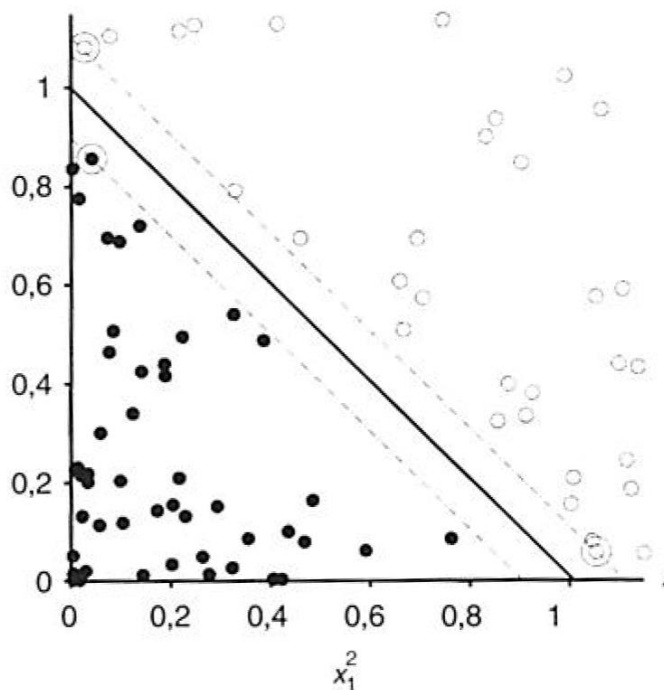


Figura 9 - Uma visão ampliada, projetada sobre as duas primeiras dimensões, do separador ótimo. O separador é mostrado como uma linha grossa, com os pontos mais próximos - os vetores de suporte - marcados com o círculo. A margem é a separação entre os exemplos positivos e negativos. (NORVING; RUSSEL, 2004, p.714)

No caso da figura 9 podemos dizer que é uma problematização linear, pois é possível separar os conjuntos por um hiperplano e o mesmo não acontece nas figuras 7 e 8 em que a problematização é não linear.

Para o entendimento de um hiperplano, e de como é feito o cálculo da distância entre hiperplanos para a separação de classes no SVM, faremos uso da explicação extraída em “Uma Introdução às Support Vector Machines” por Lorena, Ana e Carvalho André (2007).

A equação de um hiperplano é dada em (8) em que $w \cdot x$ é o produto escalar entre os vetores w e x , $w \in x$ é o vetor normal ao hiperplano descrito e $\frac{b}{||w||}$ corresponde à distância do hiperplano em relação à origem.

$$f(x) = w \cdot x + b = 0 \quad (8)$$

Essa equação divide o espaço dos dados X em duas regiões: $w \cdot x + b > 0$ e $w \cdot x + b < 0$. Uma função sinal $g(x) = \text{sgn}(f(x))$ pode então ser empregada na obtenção das classificações, conforme ilustrado na equação abaixo.

$$g(x) = \text{sgn}(f(x)) = \begin{cases} 1 & \text{se } w \cdot x + b > 0 \\ -1 & \text{se } w \cdot x + b < 0 \end{cases} \quad (9)$$

A partir de $f(x)$, é possível obter um número infinito de hiperplanos equivalentes, pela multiplicação de w e b por uma mesma constante. Define-se o hiperplano canônico em relação ao conjunto S como aquele em W e b são escalados de forma que os exemplos mais próximos ao hiperplano $w \cdot x + b = 0$ satisfaçam a equação 10.

$$|w \cdot x_1 + b| = 1 \quad (10)$$

(10) implica na inequação (11).

$$\begin{cases} w \cdot x_i + b \geq 1 & \text{se } y_i = 1 \\ w \cdot x_i + b \leq -1 & \text{se } y_i = -1 \end{cases} \quad (11)$$

Seja x_1 um ponto no hiperplano $H1: w \cdot x + b = +1$ e x_2 um ponto no hiperplano $H2: w \cdot x + b = -1$, conforme ilustrado na Figura 6. Projetando $x_1 - x_2$ na direção de w , perpendicular ao hiperplano separador $w \cdot x + b = 0$, é possível

obter a distância entre os hiperplanos H_1 e H_2 . Essa projeção é apresentada na Equação (12).

$$(x_1 - x_2) \left(\frac{w}{\|w\|} \cdot \frac{(x_1 - x_2)}{\|x_1 - x_2\|} \right) \quad (12)$$

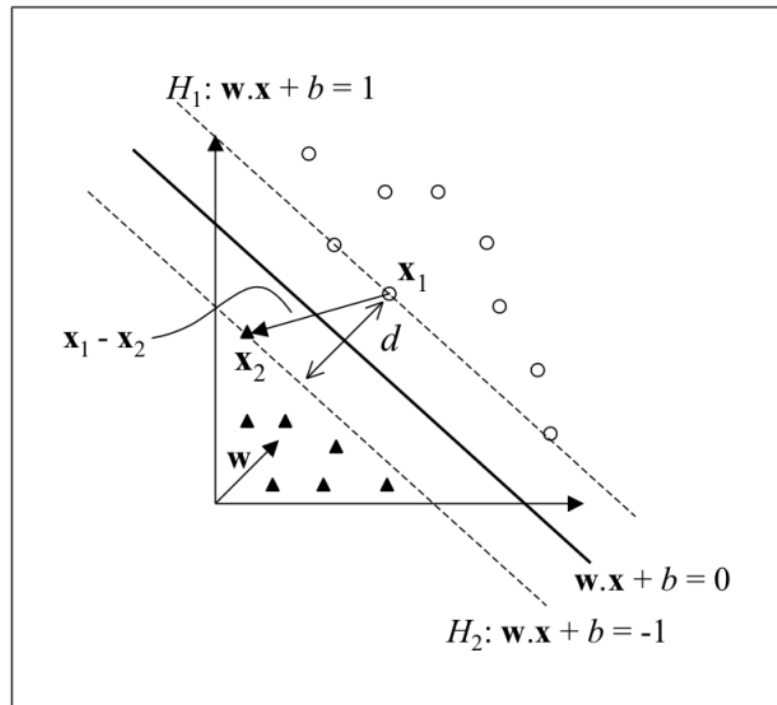


Figura 10 - Cálculo da distância d entre os hiperplanos H_1 e H_2

A distância d entre os hiperplanos H_1 e H_2 é a distância dos conjuntos de treinamento dentro do SVM. Na figura 10, é possível notar que não existe nenhum conjunto de treinamento próximo ao separador ótimo o que classifica esse SVM como linear com margens rígidas.

Podemos dizer que o SVM é um algoritmo eficiente, robusto, tem seu treinamento como simples e funciona muito bem diante de grandes dimensões de dados. Entretanto existem também algumas limitações, tais como: apresenta apenas solução binária para os conjuntos de treinamento e sensibilidade a escolhas de valores de parâmetro somado a dificuldade de interpretação do modelo gerado.

3 METODOLOGIA

Neste capítulo apresentados os métodos utilizados para o desenvolvimento deste trabalho, os materiais utilizados e os casos de teste aplicados.

3.1 MODELO DE PROCESSO DE ENGENHARIA DE SOFTWARE

3.1.1 PLANO DE ATIVIDADE

O planejamento das atividades desenvolvidas no decorrer do desenvolvimento do SBIM foi realizado utilizando as ferramentas de WBS e Gantt.

3.1.1.1 WBS

O *Work Breakdown Structure* (WBS), ou em português Estrutura Analítica de Projeto (EAP) é basicamente um checklist que identifica todas as partes do projeto e as tarefas associadas, em outras palavras, ele subdivide o projeto em partes menores, tornando mais fácil o gerenciamento. (MARTINS, 2007)

O WBS é de grande importância para o gerenciamento de um projeto, pois com base nele o escopo, tempo, custo, qualidade, recursos humanos, comunicação, riscos e aquisições do projeto são planejados.

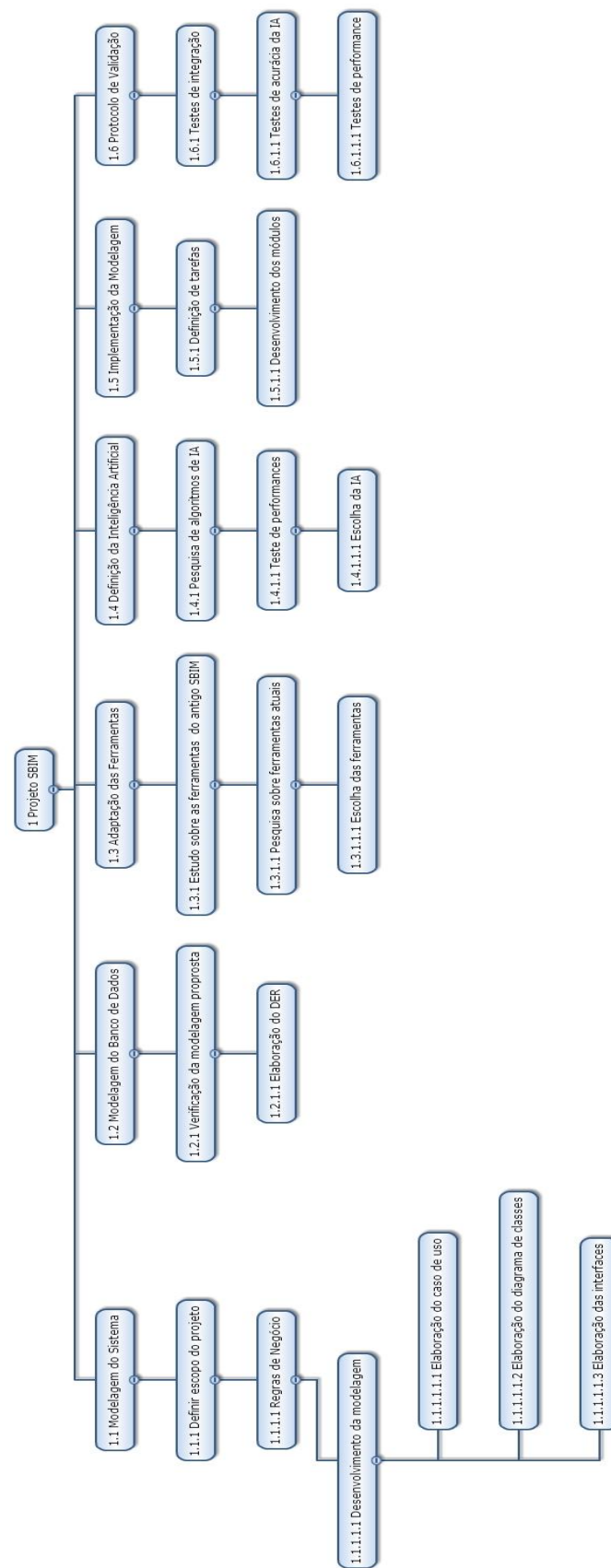


Figura 11 - WBS do SBIM

3.1.1.2 GANTT

O Diagrama de Gantt (ou Gráfico de Gantt) é utilizado para ilustrar o intervalo de tempo de cada etapa do projeto, com ele é possível ter um acompanhamento mais adequado das atividades desenvolvidas e o desempenho dos participantes do projeto.

As etapas referentes ao desenvolvimento do SBIM são:

- Planejamento
- Iniciação
- Elaboração
- Construção
- Implantação

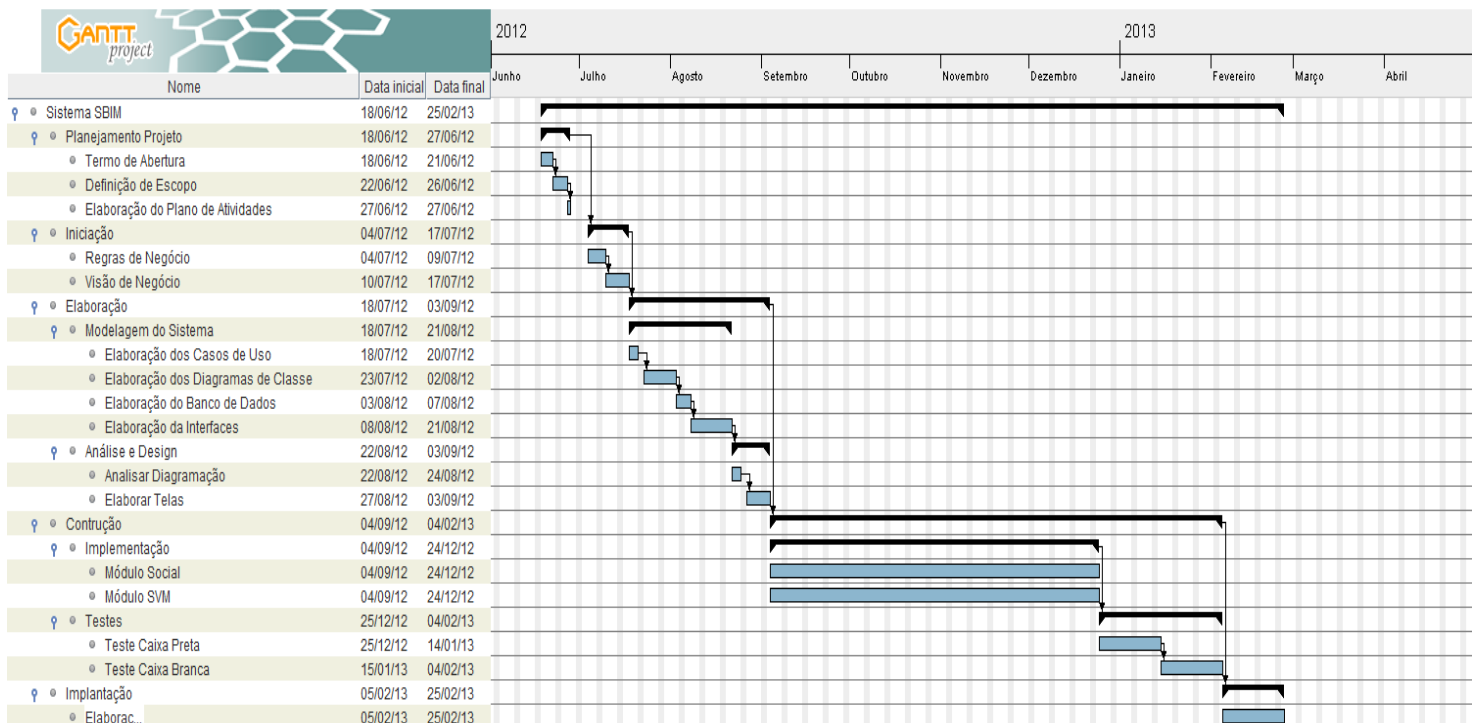


Figura 12 - Diagrama de Gantt

3.1.2 PLANO DE RISCOS

O risco de um projeto é a possibilidade de que um evento inesperado ocorra, afetando os objetivos do projeto de forma negativa ou positiva. Esta análise trata da avaliação e enfrentamento dos riscos, evitando e controlando o que for possível, minimizando a imprevisibilidade.

Os riscos encontrados no projeto foram:

- Variação do Escopo do Projeto durante o Desenvolvimento
- Falta de Tempo Hábil Para Desenvolvimento
- Perda de Dados Devido a Falha dos Equipamentos
- Desistência de Membros da Equipe Durante o Projeto
- Quantidade de Membros Insuficiente
- Inexperiência dos Membros
- Dificuldade de Interação Entre os Membros
- Problemas Pessoais

As tabelas de análise de riscos encontram-se no apêndice I.

3.2 MATERIAIS

Neste item são descritos os recursos de hardware e software utilizados para a concepção do projeto

3.2.1 AMBIENTE DE SOFTWARE

3.2.1.1 JAVASCRIPT

É uma linguagem de programação leve, interpretada e com recursos de orientação a objetos. Ele permite ter conteúdo executável do lado cliente, isto é, uma página web não precisa ser totalmente estática, ela pode incluir programas que interagem com o usuário, controlam o navegador e criam conteúdo HTML dinamicamente (FLANAGAN, 2004). Muitas interações do sistema dependem de

javascript, portanto é necessário a utilização de um navegador com amplo suporte à esta linguagem para o correto funcionamento do SBIM.

3.2.1.2 JAVA

O Java é uma linguagem interpretada, totalmente orientada a objetos, uma grande vantagem é a possibilidade de ser executada em vários sistemas operacionais diferentes, independente da implementação, contanto que seja compatível com a JVM. Ele é um sistema robusto, visando programas que precisam ser confiáveis. É de fácil programação, não é necessário se preocupar com ponteiros ou memória, porém existem maneiras de utilizar se for necessário (HORSTMANN, 2001). A versão utilizada no projeto é a 7.0.

3.2.1.3 SQL

É uma linguagem não-procedural, isto é, não é possível definir variáveis ou estruturas de laços, as instruções SQL apenas definem as entradas e saídas necessárias, a forma de execução é definida pelo mecanismo de banco de dados utilizado, decidindo o caminho de execução mais eficiente (BEAULIEU, 2010). No projeto a maior parte do código SQL é gerado pelo Hibernate.

3.2.1.4 HTML

A HTML é uma linguagem de marcação composta de elementos que definem o texto e os objetos do documento. Ela demarca o documento com notas, informando o software o que deverá ser feito em uma seção (SAVOLA, 1997). A HTML é interpretada por qualquer browser atual, desde computadores pessoais até smartphones.

3.2.1.5 APACHE TOMCAT

O *Apache Tomcat* é servidor de aplicação JEE open-source, que trabalha com a execução de Java Servlets e Java Server Pages, permitindo o funcionamento no ambiente WEB. Foi escolhido para este projeto por ser um servidor leve e

robusto, contanto com as ferramentas necessárias para a execução do SBIM (TOMCAT, 2013).

3.2.1.6 ASTAH COMMUNITY

O Astah Community é uma ferramenta grátis de modelagem UML, simples de utilizar e muito dinâmica, com os recursos necessários para a criação de vários tipos de diagramas. Esta é uma versão gratuita da ferramenta, que não contém todos os recursos, porém foi suficiente para suprir as necessidades deste projeto (ASTAH, 2013).

3.2.1.7 APACHE SUBVERSION

O *Apache Subversion* é um sistema de controle de versões open-source, com finalidade de gerenciar as diferentes versões do projeto, gerando um histórico e facilitando a integração entre o que foi produzido pelos desenvolvedores e diminuindo os problemas causados por conflitos (SUBVERSION, 2013).

3.2.1.8 JSF

O Java Server Faces é uma especificação de framework de componentes para desenvolvimento Web em Java, sendo apoiado por grandes empresas como Apache, IBM, Oracle, Siemens e outras (LUCKOW, MELO, 2012). A escolha do JSF para o projeto deve-se pela grande quantidade de recursos e informações que podem ser encontradas, tornando o desenvolvimento mais rápido e fácil. A implementação utilizada foi a Sun Mojarra que possui grande integração com o projeto Netbeans.

3.2.1.9 HIBERNATE

Utilizado para a padronização das consultas ao banco de dado, O *Hibernate* automatiza todo o processo, além de facilitar a persistência entre objetos na

aplicação e suas respectivas tabelas no banco de dados (LUCKOW, MELO, 2012). Apesar diminuir a necessidade de se fazer SQL na programação ainda é necessário ter conhecimento do que esta sendo gerado, para evitar perda de desempenho das queries (HIBERNATE, 2013).

3.2.1.10 POSTGRESQL

O *PostgreSQL* é um sistema gerenciador de banco de dados relacional de código aberto. Seu desenvolvimento perdura a mais de 15 anos, o que deu a ele boa reputação entre empresas e desenvolvedores por ser um sistema seguro e estável. Pode ser executado em vários sistemas operacionais, como Windows, Linux, Mac OS, Solaris e BSD. Seu desenvolvimento é patrocinado por empresas como Fujitsu e Red Hat (POSTGRESQL, 2013).

3.2.1.11 MYSQL WORKBENCH

O MYSQL WORKBENCH foi utilizado no projeto para a modelagem do banco de dados. Este é a evolução do antigo *DBDesigner4* e traz as ferramentas necessárias para a criação de Diagrama de Entidade e Relacionamento de maneira fácil (WORKBENCH, 2013).

3.2.1.12 LIBREOFFICE

O *LibreOffice* é uma suíte de escritório gratuita compatível com as principais suítes disponíveis no mercado atualmente, oferecendo todas as principais funcionalidades, como editor de textos, planilhas e apresentações. (LIBREOFFICE, 2013)

3.2.1.13 LIBSVM

A LIBSVM (LIBSVM, 2013) foi a biblioteca escolhida para utilizarmos o SVM no trabalho proposto. Ela tem integração com diversas linguagens e principalmente Java. Suas principais características são:

- Diferentes formulações SVM;
- Classificação multi-classe eficiente;
- Validação cruzada para seleção de modelos;
- Estimativas de probabilidades;
- Kernels diferentes (incluindo matriz Kernel pré-computadas);
- SVM ponderado para dados desbalanceados.

3.2.1.14 DROPBOX

É um serviço gratuito de armazenamento de arquivos na nuvem, que permite o compartilhamento de documentos e imagens com qualquer usuário do serviço, sendo necessário apenas a conexão com a internet (DROPBOX, 2013).

3.2.1.15 GOOGLE CODE

Utilizado para o armazenamento de código fonte, contando com ferramentas de acompanhamento de problemas, comparação de versões, *wiki* para a documentação. É integrado com o Subversion (CODE, 2013).

3.2.2 AMBIENTE DE HARDWARE

O hardware utilizado para o desenvolvimento do projeto SBIM foram as seguintes:

Notebook Lenovo Z470 com:

- Processador Intel Core i5 2ª geração;
- 4gb de memória RAM;
- Disco rígido de 500gb;
- Sistema operacional Windows 7 Home Premium 64bit.

Notebook Dell XPS com:

- Processador Intel Core i7 3ª geração;
- 4gb de memória RAM DDR3;
- Disco rígido 500gb + 32gb mSATA SSD;
- Placa gráfica NVIDIA GeForce GT 630M 1gb;
- Sistema operacional Windows 7 Ultimate 64bit;

3.3 DESENVOLVIMENTO DO PROJETO

A metodologia utilizada neste projeto foi o conteúdo visto na revisão bibliográfica juntamente com a ferramenta UML e está dividida em seis etapas descritas a seguir:

- Etapa 1 - Modelagem do sistema;
- Etapa 2 - Modelagem do banco de dados;
- Etapa 3 - Adaptação das ferramentas;
- Etapa 4 - Definição da inteligência artificial;
- Etapa 5 - Implementação da modelagem proposta;
- Etapa 6 - Protocolo de avaliação.

Estas etapas estão descritas nas próximas subseções.

3.3.1 ETAPA 1 - MODELAGEM DO SISTEMA

Nesta etapa apresentamos as funcionalidades do sistema.

3.3.1.1 DIAGRAMA DE CASOS DE USO

O diagrama de casos de uso tem o objetivo de mostrar todas as funcionalidades do sistema com os seus respectivos utilizadores, assim, em uma só imagem o cliente terá uma visão geral do sistema que será implementado, evitando problemas de falta de componentes durante o ciclo de desenvolvimento do projeto.

Os diagramas de Casos de Uso do SBIM encontram-se no apêndice A, junto com suas respectivas especificações.

3.3.1.1.1 CASO DE USO DO SBIM

O caso de uso do SBIM mostra a interação que o usuário pode ter com o sistema. Um usuário pode se cadastrar no SBIM (Realizar Cadastro), editar os seus dados pessoais (Editar Dados Cadastrais), buscar outros pesquisadores que fazem parte do SBIM (Buscar Usuários), enviar mensagens (Enviar Mensagem), adicioná-los como contatos (Adicionar Contatos), ou então convidá-los para participar de bases (Convidar Contatos Para Bases), visualizar os convites para base (Visualizar Convites Para Base) e para contato (Visualizar Convite Para Contato), também é possível criar bases de dados (Criar Nova Base), e após adicionar imagens (Adicionar Imagem a Base) é possível criar classes (Criar Classe), classificar as imagens (Classificar Imagens), aplicar efeitos a imagens (Adicionar Efeito na Imagem), realizar marcações (Realizar Marcações na Imagem), e depois com os dados retirados de todas as imagens presentes no sistema é possível fazer upload de uma imagem e buscar aquelas que forem mais semelhantes (Buscar Imagens Semelhantes).

3.3.1.2 DIAGRAMA DE CLASSES

A função principal do diagrama de classes é expor o conjunto de classes existente, suas interfaces, o relacionamento entre seus componentes (BOOCH, RUMBAUGH, JACOBSON, 2000) e determinar os atributos e métodos possuídos por cada classe fornecendo uma visão geral do sistema. Este diagrama tem uma grande importância no UML, pois serve de referência para muitos outros diagramas.

O diagrama de classes do sistema encontra-se no apêndice D.

3.3.1.3 DIAGRAMA DE SEQUÊNCIA

O diagrama de sequência representam as ações que ocorrerão no sistema durante a execução de um caso de uso, ou seja, mostrar a interação dos objetos ao longo do tempo e as mensagens trocadas entre eles.

Os diagramas de sequência encontram-se no apêndice F.

3.3.1.4 DIAGRAMAS DE SEQUÊNCIA DO SBIM

a) Cadastro de Usuário

Qualquer pessoa pode se cadastrar no SBIM, para isso é necessário preencher o cadastro disponível na página inicial do sistema. Para efetivar o cadastro os campos “*First Name*”, “*Last Name*”, “*Identification*”, “*Email*”, “*Confirm Email*”, “*Password*” e “*Confirm Password*” devem ser preenchidos, sendo que o campo “*Email*” é o mesmo de “*Confirm Email*” e “*Password*” é o mesmo de “*Confirm Password*”. Assim tendo certeza que é o e-mail e a senha que o usuário deseja.

b) Edição de Dados

Se o usuário não preencheu algum dado não obrigatório na hora do cadastro ou deseja modificar algo do seu atual cadastro, ele pode alterar os dados desejados na página de edição.

c) Busca de Usuário

Sempre que necessário o usuário logado poderá buscar outros usuários, seja para verificar suas últimas atualizações, visualizar suas bases ou contatos, ou então convidá-lo para participar de bases ou tornar-se contato. A cada letra digitada no campo de busca o componente autocomplete faz uma chamada ao SearchBean, ele que faz uma chamada à regra de negócio, que faz uma chamada ao Hibernate, retornando uma lista de usuários que o nome corresponde com o parâmetro digitado no campo busca. Na tela é mostrado uma lista, o usuário seleciona o perfil de quem ele está procurando e é redirecionado a página requisitada.

d) Nova Base

Todo usuário pode criar novas bases de dados no SBIM, uma criação de base funciona da seguinte maneira: ao abrir a página o sistema carrega o usuário que está logado, pegando o e-mail que está na sessão, carregando-o e deixando disponível para acesso posterior, após isso as áreas disponíveis são listadas, montando o combobox “áreas”. Depois que a página é carregada o usuário digita o nome da sua nova base, seleciona a área referente a ela e clica no botão salvar,

após esse passo o sistema relaciona a base ao usuário que a criou (o usuário logado) e a salva no banco de dados.

e) Adicionar Imagem a Base

A principal funcionalidade da base é a possibilidade de adicionar imagens a ela, dando a todos os participantes da base a possibilidade de visualizar, editar e comentar as imagens, assim demonstrando detalhes que passaram despercebidos antes por outros usuários. Apenas as imagens nos formatos JPG, GIF, PNG e BMP são suportados.

f) Adicionar Contato

Os usuários do SBIM a fim de aumentar sua rede de contatos podem enviar convites a outros usuários, tornando mais fácil a comunicação entre os pesquisadores e o convite para a participação de bases.

g) Adicionar Contatos Para Base

Assim como é possível adicionar contatos a sua lista também é possível convidá-los a participar as suas bases, tornando mais fácil o compartilhamento de imagens e informações. Este processo ocorre da seguinte maneira: ao entrar no perfil do usuário que se deseja fazer o convite é chamada a classe InviteBean, que carrega os dados do usuário que ira fazer o convite, junto com as suas bases, e os dados do usuário que irá receber o convite. O usuário deverá clicar no botão 'Invite to a group', ele abrira uma janela com as bases disponíveis para a realização do convite, o usuário clica no botão 'Invite' da base desejada, é chamado o método 'InviteGroup' que cria uma nova instancia do objeto 'Invitations', os dados dos usuários são setados e o novo convite é salvo no banco de dados.

h) Marcação de Imagem

Depois que é feito o upload da imagem em uma base é possível fazer marcações nela, para exaltar alguns detalhes que poderão ser discutido entre os pesquisadores. A maior parte da marcação acontece graças a funções JavaScript executadas no navegador do cliente, tornado o sistema mais interativo. Para marcar uma imagem é necessário selecionar a área desejada, ao finalizar essa seleção é chamada uma função JavaScript que mostrará um input de texto, que deverá ser

colocado o nome que será dado a aquela tag, ao teclar “Enter” é chamada a função “FinalizaMark”, que destroi o input e cria um componente HTML 'span' na imagem, exibindo assim a marcação finalizada. Durante esta mesma função a classe 'ImageBean' é chamada, criando uma nova instância do objeto 'Marking' e o adicionando a lista de marcações da imagem editada. Para salvar esta marcação o usuário deverá clicar no link 'Save Edits'.

j) Busca de Imagem

No SBIM é possível fazer busca de imagens, ao fazer o upload de uma imagem o sistema pesquisará as imagens que mais se aproximam desta. Este processo ocorre da seguinte forma: ao selecionar a imagem e clicar no botão 'upload' o sistema irá extrair as características da imagem enviada, logo após será chamada a função 'prediction', que procura a classe com as imagens que mais se assemelham a aquela em que foi feito o upload. Depois que a classe é definida as imagens pertencentes a ela são exibidas para o usuário.

j) Classificação de Imagem

Para armazenar as características de uma imagem e organizá-las pelo mesmo tipo é preciso classificá-las. Para que este processo ocorra é necessário o usuário estar na página de classificação, nela são apresentadas as imagens referentes a uma base. O usuário seleciona as imagens que deverão ser classificadas e escolhe a classe, neste instante o sistema extrai as características de todas as imagens e as salva em uma pasta específica e a classe é relacionada à imagem.

3.3.1.5 DIAGRAMA DE ATIVIDADES

Basicamente um diagrama de atividades é um gráfico de fluxo. O objetivo é mostrar o fluxo de controle de uma atividade para outra, em outras palavras, este diagrama preocupa-se em descrever os passos a serem percorridos para a conclusão de uma atividade e concentra-se na representação do fluxo de controle de uma atividade. (GUEDES, 2004)

3.3.2 ETAPA 2 - MODELAGEM DO BANCO DE DADOS

Nesta etapa é definida a modelagem do banco de dados do SBIM, de maneira que todas as especificações apresentadas na etapa anterior sejam atendidas. Para esta modelagem foi utilizada a ferramenta MySQL Workbench e o SGBD escolhido foi o PostgreSQL. Os Relacionamentos entre as tabelas do banco de dados são ilustrados no apêndice E.

Dentre as tabelas utilizadas no sistema, as principais são:

- Tabela “bases”: utilizada para armazenar os dados das bases criadas pelos usuários do sistema.
- Tabela “images”: representa os dados das imagens adicionadas ao sistema, contendo informações como tamanho, compressão e dados das características da imagem.
- Tabela “invitations”: contém os dados de convites enviados, podendo ser referente a contatos e bases.
- Tabela “svm_class”: é utilizada para armazenar os dados das classes presentes no sistema, armazenando os dados de redução e classificação.

O dicionário de dados é mostrado no apêndice G.

3.3.3 ETAPA 3 - ADAPTAÇÕES DAS FERRAMENTAS

No Antigo SBIM (Neves, Porcides, Giraldi, 2011) a linguagem utilizada é o PHP. Apesar de sua aprendizagem e utilização seja mais fácil no desenvolvimento de pequenos websites, no Novo SBIM foi necessário o uso de ferramentas mais robustas e com mais recursos de edição de imagens. Como solução foi escolhida a linguagem Java juntamente com alguns frameworks adjacentes a ela tais como JavaServer Faces, *Spring Security* e *Hibernate* para o desenvolvimento do novo sistema proposto. A linguagem escolhida, além de conter mais recursos, é totalmente orientada a objetos o que facilita a manutenabilidade do sistema, que foi inclusive uma das problematizações deste trabalho.

3.3.4 ETAPA 4- USO DA INTELIGÊNCIA ARTIFICIAL

Na revisão bibliográfica, tivemos uma introdução teórica do trabalho proposto, e nesta seção temos a explicação de como foi feito o uso da inteligência artificial no SBIM para que pudéssemos ter a classificação e recuperação de imagens através de conteúdo. Esta etapa é dividida em quatro fases. São elas:

- Criação e população das bases com imagens;
- Extração de características;
- Treinamento das classes;
- Recuperação de imagens.

3.3.4.1 CRIAÇÃO DE BASE

Nesta fase o usuário cria uma base de estudo e convida alguns de seus contatos os quais deseja compartilhar os estudos. A partir deste momento, todos os participantes da base podem fazer *upload* de imagens.

3.3.4.2 EXTRAÇÃO DE CARACTERÍSTICAS

Após a criação de uma base de estudos, o usuário pode acionar a classificação das imagens inseridas na base. Quando acionada, o sistema extrai as características de cada uma das imagens da base.

Na classificação, para cada imagem, são aplicados cinco passos:

1. Redimensionar imagem para 128 x 128.
2. Extração dos valores de RGB da imagem;
3. Aplicação da Transformada discreta do cosseno no vetor que representa a imagem;
4. Eliminar as redundâncias da imagem;
5. Armazenamento do arquivo de características no banco de dados.

No primeiro passo, redimensionamos a imagem para 128 x 128 - utilizando interpolação bi linear – para que haja um padrão entre as imagens inseridas na base

e para diminuirmos o custo computacional. Se pensarmos que podem existir imagens com grande resolução, isso já seria um ganho considerável.

No segundo passo, carregamos a imagem para a memória e inserimos em um vetor bidimensional V os valores de RGB da imagem.

No terceiro passo, aplicamos a transformada discreta do cosseno (DCT) no vetor V e obtemos como saída um novo vetor T de mesma dimensão, com os valores da transformada.

No quarto passo, eliminamos a redundância da imagem desconsiderando uma parte dos coeficientes obtidos no cálculo da DCT. Portanto, representamos uma imagem considerando um novo universo de 40x40 do vetor transformado e armazenamos os valores em um arquivo de extensão “fet”.

No último passo, após a criação do arquivo que contém as características da imagem, armazenamos o arquivo no banco de dados.

3.3.4.3 TREINAMENTO

Para o treinamento das classes, utilizamos uma técnica de aproximação chamada de “*one-against-all*”. Essa técnica consiste em comparar cada classe do sistema contra as demais. Se no sistema existir dez classes, cada classe será comparada contra nove classes. Comparar uma classe significa criar um arquivo de treinamento no qual todas as imagens da classe X terão *label* 1 e um conjunto de imagens das demais classes como -1. Se na classe X existem K imagens, em seu conjunto de treinamento haverá uma seleção de $2K$ imagens das outras classes em que cada uma delas terá $\frac{2K}{(N-1)}$ imagens sendo N o número total de classes do sistema.

Pensando na aproximação “one-against-all” e analisando a figura 9, poderíamos considerar que o conjunto de bolas pretas representa a classe X e o conjunto de bolas brancas representa a seleção de $2k$ imagens das demais classes do sistema. A partir disso, o SVM vai buscar o separador ótimo entre os dois hiperplanos. Depois de prepararmos o arquivo para treinamento, treinamos nossa

classe e criamos um novo arquivo chamado de “model”. Seu caminho é armazenado no banco de dados.

3.3.4.4 RECUPERAÇÃO DE IMAGEM

Nesta fase, o sistema recupera do banco de dados às imagens mais próximas daquela que foi passada como consulta, que nesse caso vamos chama-la de “imagem consulta”. Inicialmente, aplicamos as mesmas técnicas descritas na seção 3.3.4.2 na “imagem consulta” para extrairmos suas características, entretanto, armazenamos o arquivo “fet” em uma pasta temporária.

No segundo momento, recuperamos do banco de dados todas as classes que o sistema possui e para cada uma delas o SVM nos retorna um valor entre -1 e +1 em que, quanto mais próximo de 1 for o valor, maior a chance da “imagem consulta” pertencer a dada classe. A classe que obtiver o maior grau de pertinência, consideramos que as imagens desta, são as mais próximas da “imagem consulta” e neste caso calculamos a distância euclidiana entre a “imagem consulta” e todas as imagens da classe com o maior grau. Em (13) temos a fórmula da distância euclidiana para n-dimensional.

$$d = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (13)$$

O sistema retornará em forma crescente as imagens da classe com maior grau, ordenadas pela distancia euclidiana.

3.3.5 ETAPA 5 - IMPLEMENTAÇÃO DA MODELAGEM PROPOSTA

Nesta etapa é que acontece o desenvolvimento do SBIM seguindo o modelo que foi proposto nas etapas anteriores do projeto. O sistema foi desenvolvido utilizando a linguagem Java, junto com o framework JSF e o framework de componente de interface de usuários RichFaces, o sistema gerenciador de banco de dados utilizado é o PostgreSQL. No sistema foi utilizado o subversion, para

gerenciar o versionamento e o Google Code como repositório para armazenar o código fonte desenvolvido.

O trabalho foi dividido da seguinte forma:

Tabela 1 -Tabela de divisão de tarefas

	Fellipe	Douglas
Desenvolver planejamento	X	X
Elaborar os diagramas de caso de uso	X	
Elaborar o banco de dados		X
Elaborar os diagramas da UML	X	
Elaborar interfaces e telas		X
Analisar diagramação e interfaces	X	X
Implementar módulo social	X	
Implementar módulo SVM		X
Integrar os módulos		X
Desenvolver e executar casos de teste		X

Dentre os casos de risco listados na seção 3.1.2, os seguintes ocorreram neste trabalho:

- Dificuldade da Implementação do SVM

- Desistência de Membros da Equipe Durante o Projeto
- Quantidade de Membros Insuficiente
- Variação do Escopo do Projeto Durante o Desenvolvimento

Durante o mês de janeiro um membro do grupo desistiu do projeto, deixando os outros membros sobrecarregados, muitas atividades tiveram que ser repensadas, realocadas e até mesmo removidas, mas o desenvolvimento prosseguiu e pôde ser concluído mesmo com atrasos.

Como foi dito anteriormente, com a saída de um dos membros do projeto o grupo ficou desfalcado. Como o tempo estava se esgotando, algumas partes do projeto já estavam em um nível avançado e não conseguimos encontrar outra pessoa que tivesse os conhecimentos necessários para acompanhar o desenvolvimento foi decidido que o projeto iria ser finalizado com apenas dois integrantes.

O escopo do projeto teve que ser modificado algumas vezes, devido a falta de membros dedicados ao projeto, sendo assim algumas funcionalidades do sistema foram removidas e algumas outras modificadas

E os seguintes riscos que foram evitados:

- Falta de Tempo Hábil Para o Desenvolvimento
- Perda de Dados Devido a Falha dos Equipamentos
- Inexperiência dos Membros

Apesar do desfalque no grupo, atrasos e alterações do escopo o projeto pode ser finalizado com sucesso.

Para se evitar perda de dados foram utilizadas as seguintes ferramentas: Google Code (seção 3.2.1.15) e Subversion (seção 3.2.1.7) para versionamento e armazenamento do código fonte, e para armazenar os outros arquivos presentes no sistema foi utilizado o Dropbox (seção 3.2.1.14), guardando todo o necessário na nuvem.

As ferramentas escolhidas para o desenvolvimento do projeto foram escolhidas de acordo com o conhecimento dos participantes do projeto, evitando o baixo desempenho das tarefas devido a falta de conhecimento técnico. Trazendo assim mais agilidade no desenvolvimento do projeto.

O sistema foi desenvolvido como descrito no WBS, encontrado na seção 3.1.1.1, seguindo todas as etapas propostas. Também foi tomado como base do diagrama de Gantt, na seção 3.1.1.2, porém alguns atrasos ocorreram, porém não acarretou muito impacto a versão final do produto desenvolvido.

3.3.6 ETAPA 6 - PROTOCOLO DE VALIDAÇÃO

A validação do sistema é realizada em duas etapas. Na primeira delas é feito um caso de teste para cada caso de uso do sistema. Cada caso de teste contém as pré-condições para que a ação ocorra no sistema, o caminho que o usuário deve percorrer para a funcionalidade, uma lista de entrada de dados que o usuário deve inserir quando necessário, a ação para acionar o próximo passo da funcionalidade e o resultado esperado do sistema. Essa metodologia de testes é conhecida como “teste de integração”, conforme Pressman (PRESSMAN, 1995). Na segunda etapa validamos a acurácia do algoritmo Support Vector Machine na recuperação de imagens. Para tanto, fizemos uma seleção de 440 imagens e as dividimos em 4 classes. Utilizamos 90% da seleção para treinamento e os outros 10%, para teste. Na terceira etapa a velocidade da classificação das imagens e do treinamento das classes são testadas

3.3.6.1 RESULTADOS

Os testes de integração foram feitos seguindo o fluxo ideal, testando as principais funcionalidades do sistema e desconsiderando as condições de erro.

Os resultados apresentados pelo teste foram satisfatórios, demonstrando que todas as funcionalidades do sistema funcionam como o esperado. Os testes de integração encontram-se no apêndice J.

A seguir a tabela 2 mostra a precisão do SVM na recuperação das imagens utilizando as técnicas propostas neste trabalho. A acurácia é medida dividindo o número de acertos pelo total de imagens utilizado para testar a classe.

Tabela 2 -Tabela de acurácia do SVM

Classe	Imagens	Acertos	Acurácia
Face	10	10	100,00%
Iris	10	10	100,00%
Flores	10	9	90,00%
Palma da Mão	10	10	100,00%
Total	40	39	97,50%

As bases da Face, Iris e Palma da Mão apresentaram acurácia de 100%, enquanto a base das Flores obteve 90%. Isso se deve ao fato de uma das imagens utilizada para testar a classe Flores, tinha uma grande quantidade de preto no fundo. Isso fez com que o SVM retornasse um grau de pertinência muito próxima a classe “Palma da mão” que possui todas as suas imagens com fundo preto. Nas figuras 13, 14, 15 e 16 são apresentadas algumas imagens das bases que foram utilizadas para teste.



Figura 13 - Exemplo de imagens da base face

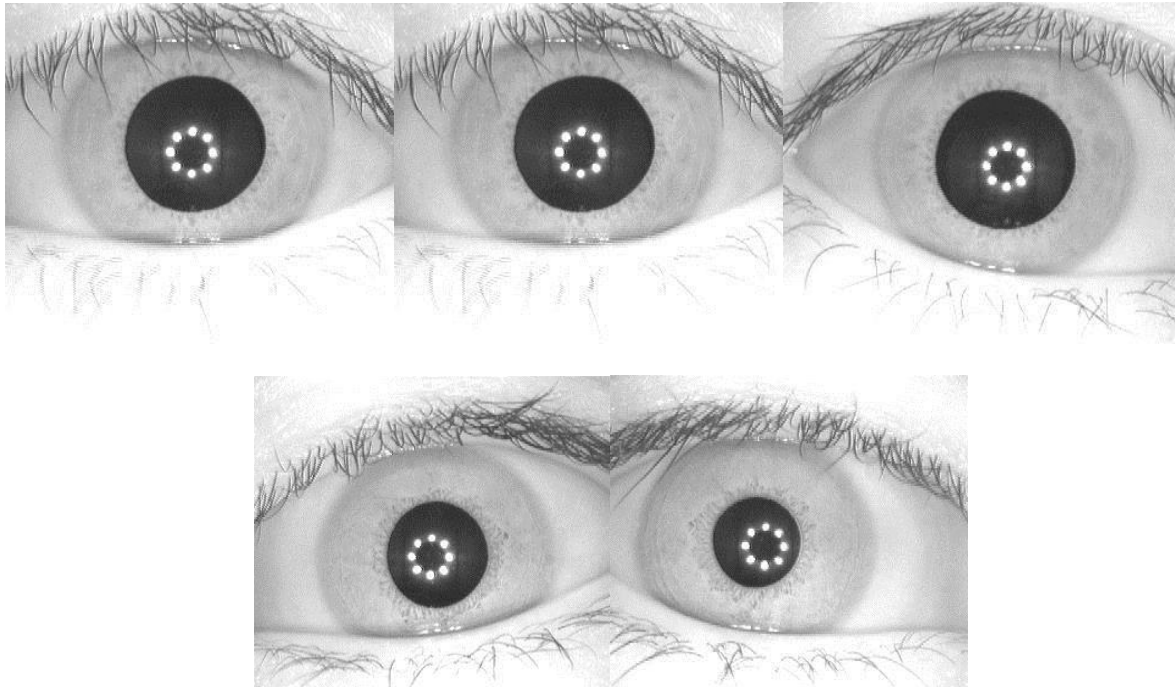


Figura 14 - Exemplo de imagens da base íris



Figura 15 - Exemplo de imagens da base flores

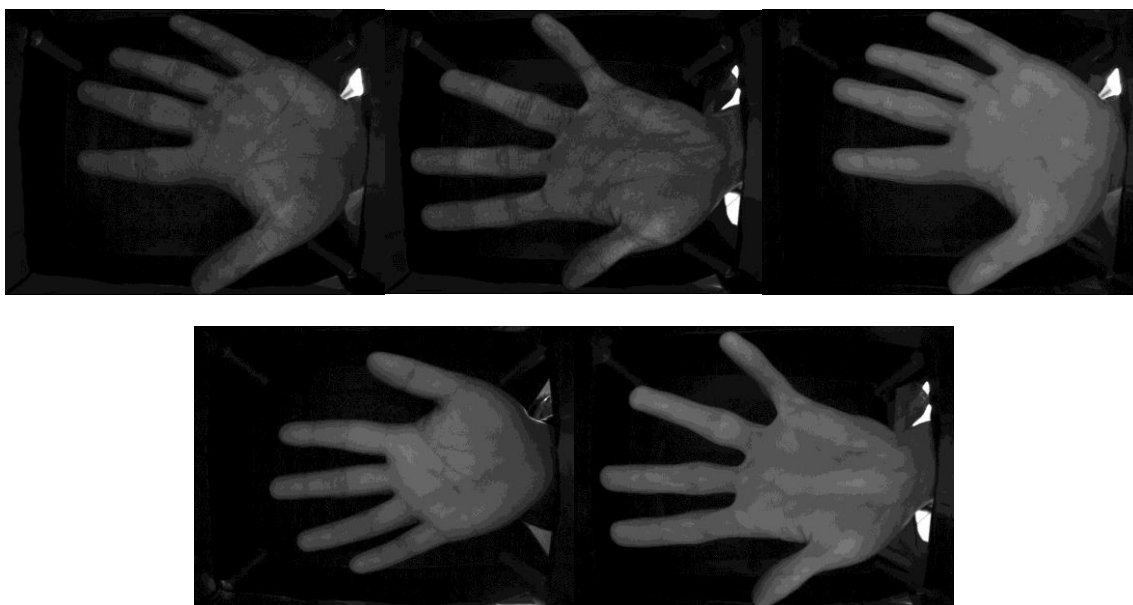


Figura 16 - Exemplo de imagens da base palma da mão

A tabela 3 mostra o tempo médio de classificação e treinamento de uma certa quantidade de imagens. Para a realização dos testes foram selecionadas uma certa quantidade de imagens, que foram classificadas e treinadas cinco vezes seguidas. A média de tempo foi definida somando os resultados obtidos e dividindo-os pelo número de testes executados

Tabela 3 - Tempos de classificação e treinamento

Número de Imagens	Média Tempo de Classificação	Média Tempo de Treinamento
10	0,374s	0,252s
20	0,699s	0,332s

Os resultados obtidos através dos testes foram satisfatórios, mostrando que a rede social teve o desempenho esperado, o SVM conseguiu recuperar imagens com uma taxa de acerto de 97,5% , além de classificar e treinar imagens de uma forma rápida.

4 APRESENTAÇÃO DO SOFTWARE

4.1 TELAS DO SISTEMA

Nesta seção serão apresentadas as telas do SBIM, junto com suas descrições.

a) REGISTRO DO USUÁRIO

Para registrar um novo usuário é necessário executar dois passos. O primeiro passo é mostrado na figura 17, nele o usuário digita os seus dados pessoais (nome, identificação), seus dados de acesso (e-mail, senha) e poderá fazer upload de uma imagem. Ao clicar no botão “Next Step” ele será encaminhado para o próximo passo do registro.

SBIM

First Name: Jose

Last Name: Souza


Identification: CPF 111111111111

E-mail: jose.souza@email.com

Confirm E-mail: jose.souza@email.com

Password: *****

Confirm Password: *****

Photo: 

[Next Step](#)

2013 SBIM - Shared Biological Image Manager


 [About](#) | [Help](#) | [Terms of Service](#) | [Privacy Policy](#)

Figura 17 - Registro passo 1

Neste passo é solicitada a sua instituição, a área de interesse, o telefone e os dados de localização, mas nenhum dos dados desse passo são obrigatórios.

SBIM

Institution: UFPR

Area of Interest: Qualquer

Phone: 33333333

Country: Brazil

City: Curitiba

Zip Code: 88888888

[Previous Page](#) [Finalizar](#)

2013 SBIM - Shared Biological Image Manager

SBIM [About](#) | [Help](#) | [Terms of Service](#) | [Privacy Policy](#)

localhost:8084/NovoSbim/index.jsf

Figura 18 - Registro passo 2

b) AUTENTICAÇÃO DO USUÁRIO

Para ter acesso ao sistema é necessário antes passar pela autenticação. A tela é representada na figura 19.

SBIM

Email

Password

[Log In](#)

Join Us

First Name

Last Name

E-mail

[Join Now](#)

What is SBIM?
The Shared Biological Image Manager aims to allow an easy way to share and manipulate images between researchers by providing access to medical and biological images and marking tools. The SBIM is a Scientific Initiation Project that aims to create an on-line medical and biological image manager. In this project, the images are separated in several bases. Here you can send messages to other researchers, recommend images and use the marking tool to make markings on the images. This project has been developed with the support of the Araucária Foundation, CNPq and the Federal University of Paraná.

Developed By
Douglas Natan Meireles Cardoso
Felipe Alexandre

2013 SBIM - Shared Biological Image Manager

SBIM [About](#) | [Help](#) | [Terms of Service](#) | [Privacy Policy](#)

Figura 19 - Autenticação do usuário

c) TELA PRINCIPAL

Depois de feito o login, o usuário será redirecionado para a página inicial do sistema. Nesta tela é exibida os contatos, as bases, as informações e as última atualizações dos contatos do usuário logado. Nesta tela, assim como todas as

outras do sistema, o usuário poderá acessar as principais funcionalidades do SBIM através do menu superior. A tela é mostrada na figura 20.

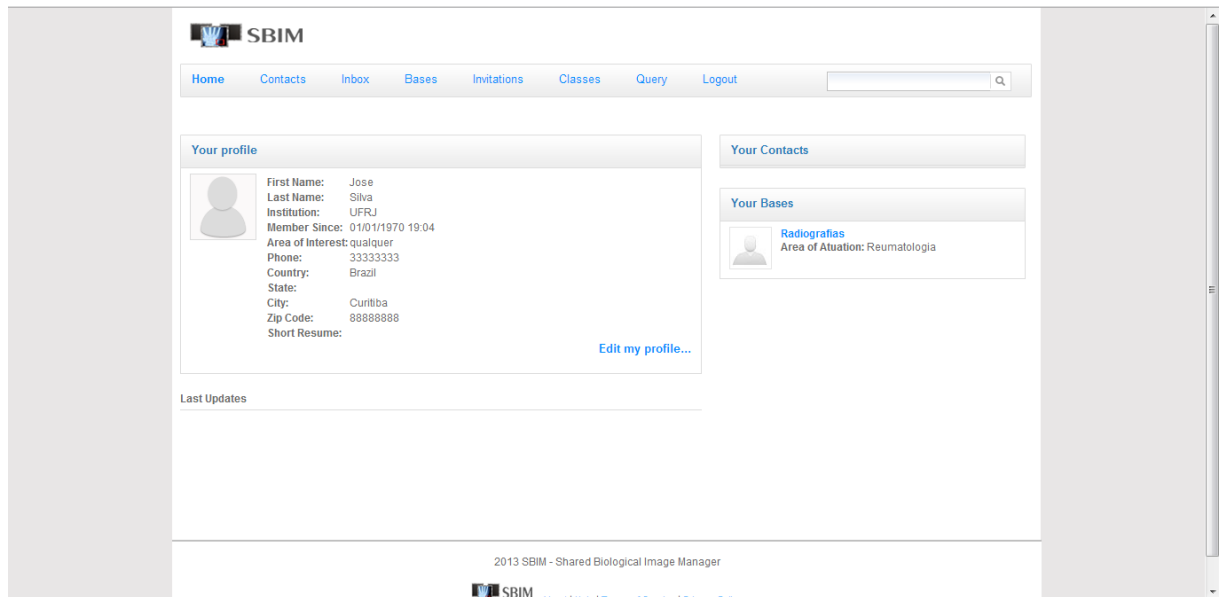


Figura 20 - Tela principal

d) ENVIO DE MENSAGENS

É exibida a tela de conversa ao clicar no nome do usuário, ao lado esquerdo. Nela são exibidas as últimas mensagens trocadas entre o usuário logado e o usuário selecionado. Esta tela é ilustrada na imagem 18.

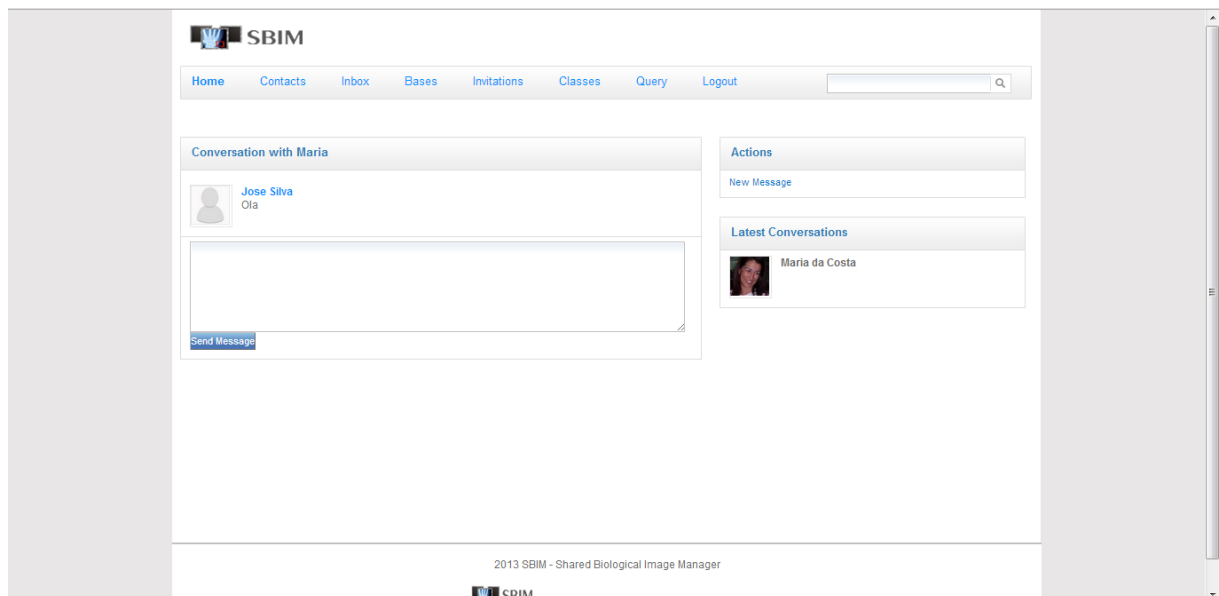


Figura 21 - Envio de mensagens

e) PERFIL DE PESQUISADORES

Esta tela é muito semelhante a tela principal, mostrando as bases e os contatos do usuário, porém mostra as últimas atualizações do usuário que está sendo visualizado. Também são exibidos os botões “Add to Contacts” (será exibido “Already a Contact” se os usuários já forem contatos) e o botão “Invite to a Base”.

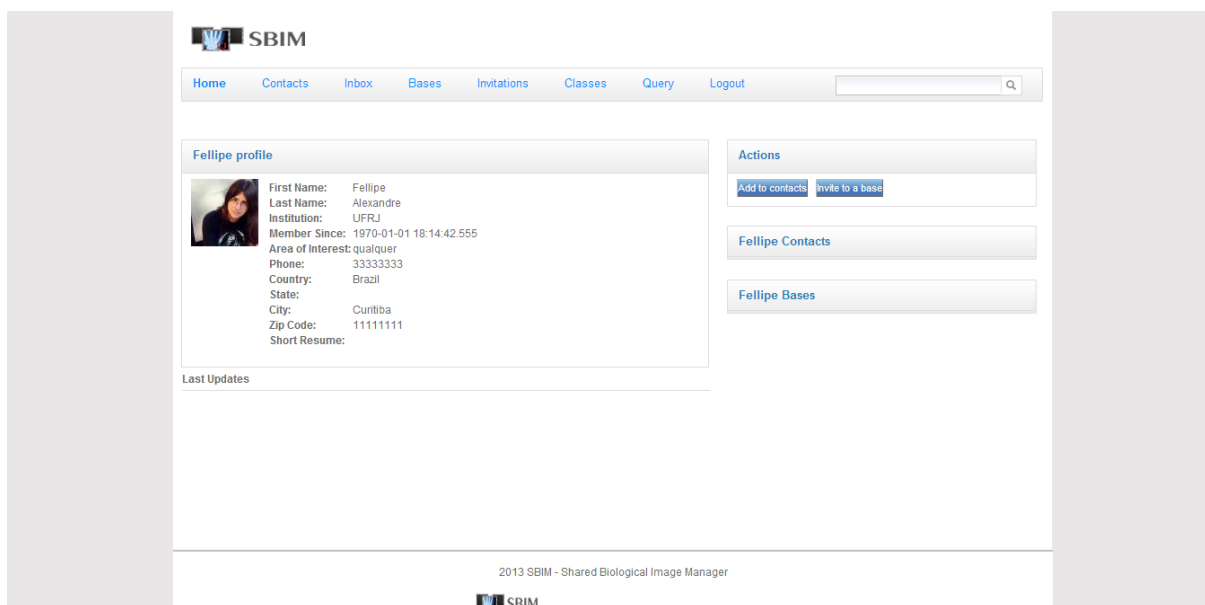


Figura 22 - Perfil do pesquisador

f) LISTAGEM DE CONVITES

Nesta tela são exibidos os convites (de contatos e bases) enviados, recebidos e aguardando aprovação. Esta tela é exibida na figura 23.

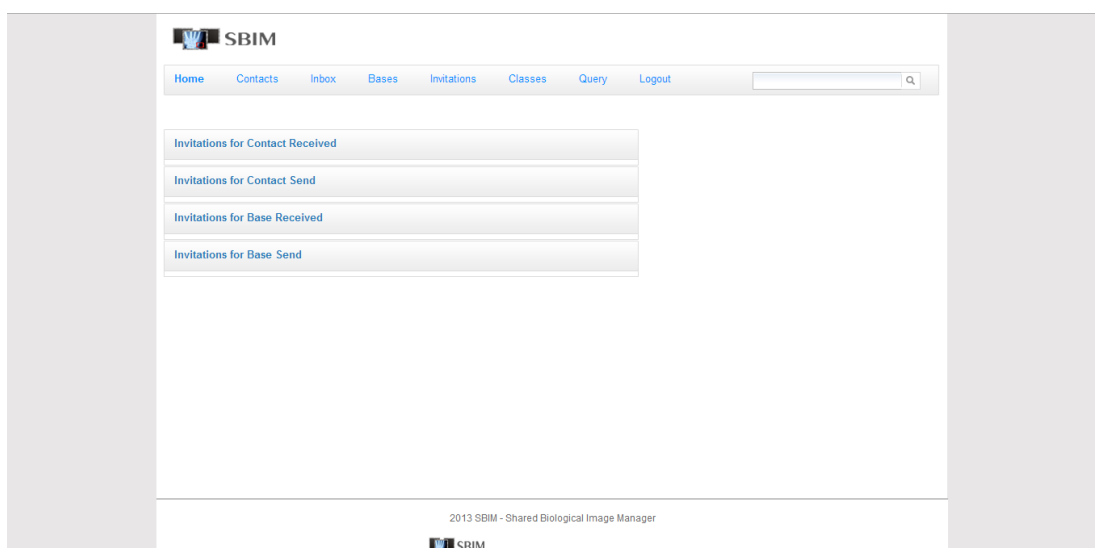


Figura 23 - Listagem de convites

g) LISTAGEM DE CONTATOS

Exibe todos os contatos do usuário logado, podendo removê-los se for requisitado. Esta tela é exibida na figura 24.

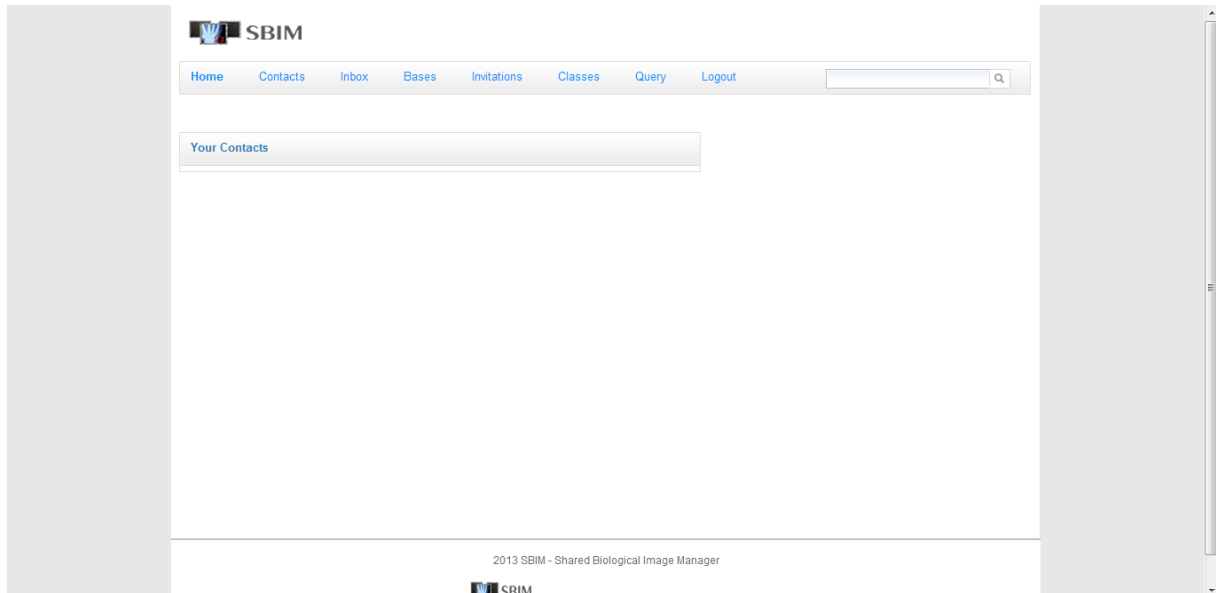


Figura 24 - Listagem de contatos

h) LISTAGEM DE BASES

Exibe as bases que o usuário logado participa. Ilustrado na figura 25.

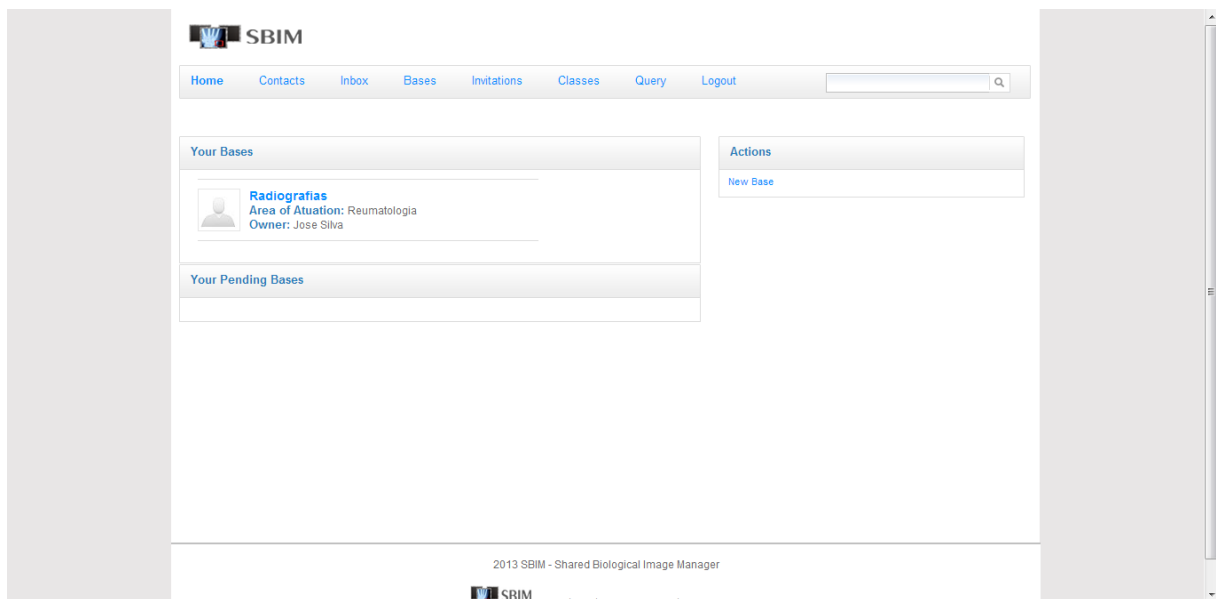


Figura 25 - Listagem de bases

i) GERENCIAMENTO DE BASES

Nesta tela as imagens presentes na base são exibidas, assim como os dados da base. Também é exibida a ferramenta de upload de imagem. Ilustrados na figura 26.

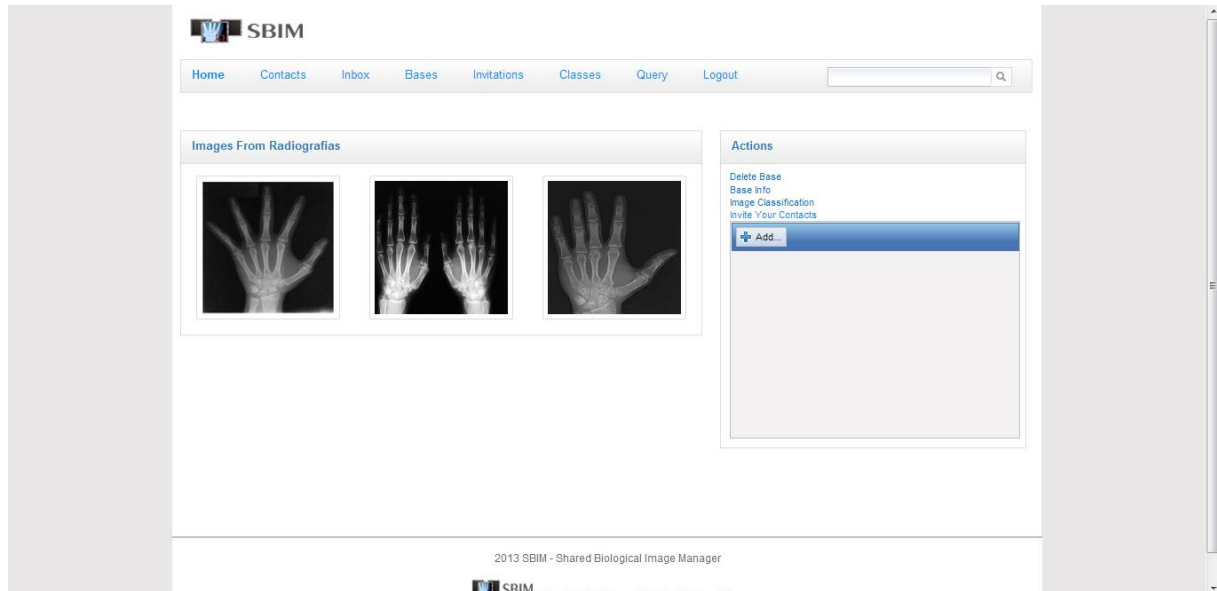


Figura 26 - Gerenciamento de base

j) EDIÇÃO DE IMAGEM

Esta tela é utilizada quando é necessário visualizar ou editar uma imagem. Nos menus laterais ficam localizadas as ações que podem ser tomadas, os efeitos que podem ser aplicados na imagem e as marcações presentes na mesma. Abaixo da imagem ficam localizados os comentários feitos relacionados a imagem. Esta tela é exibida na figura 27.



Figura 27 - Edição de imagem

k) CLASSIFICAÇÃO DE IMAGEM

As imagens presentes nas bases podem ser classificadas, acessando a tela de classificação de imagem. Para realizar a classificação é necessário selecionar as imagens e selecionar a classe desejada. Após a classificação é exibida a classe a qual pertence cada imagem.

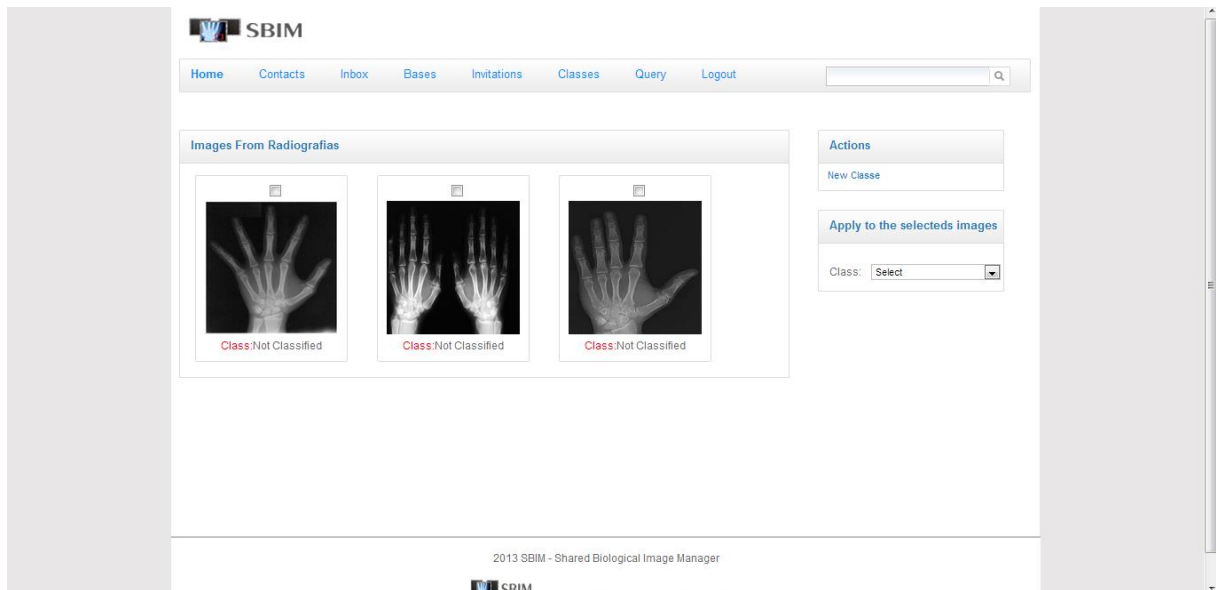


Figura 28 - Classificação de imagem

I) BUSCA DE IMAGEM

Nesta tela o usuário poderá fazer o upload de uma imagem, ela que sera comparada com as imagens registradas no sistema, retornando uma lista com as imagens consideradas mais semelhantes como mostra na figura 29.

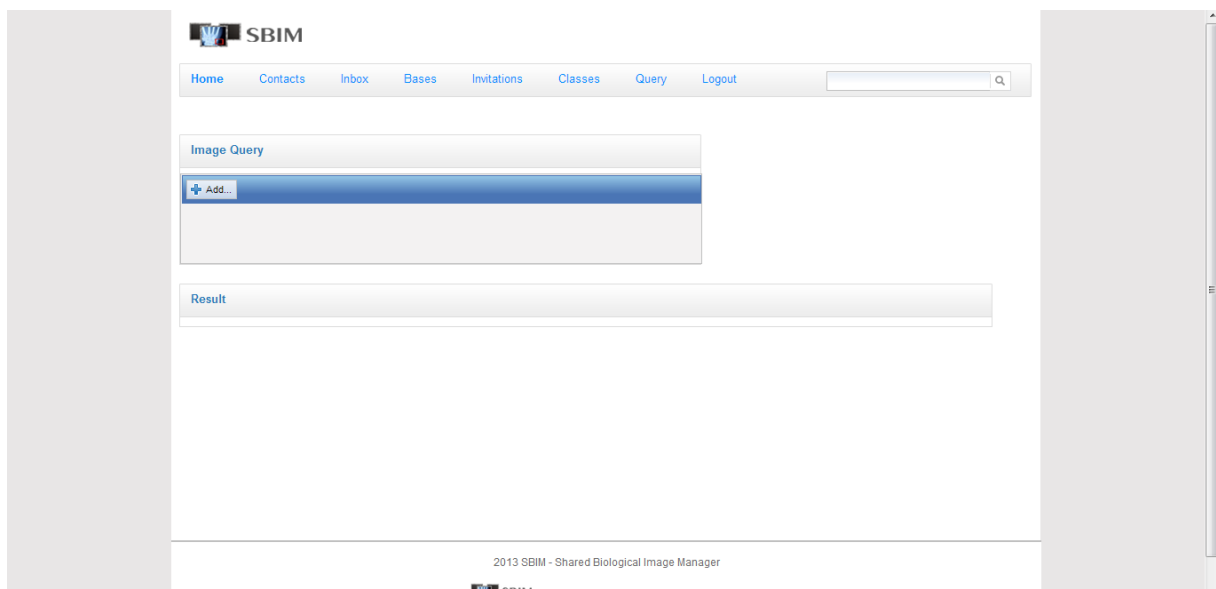


Figura 29 - Busca de imagem

4.2 MANUAL DE INSTALAÇÃO

Neste capítulo será exibido o manual de instalação do SBIM . Os pré requisitos são:

- Possuir o banco de dados PostgreSQL;
- Possuir o servidor Apache Tomcat;
- Utilizar o browser Google Chrome;
- Possuir o cd com o código fonte e script do SBIM.

Para o funcionamento do sistema os seguintes passos devem ser seguidos:

4.2.1 PASSO 1 – CONFIGURAR O BANCO DE DADOS

No PostgreSQL é necessário criar uma nova *database* chamada 'sbim'. A imagem abaixo mostra como ficará a janela de criação de *database* no PGAdmin.

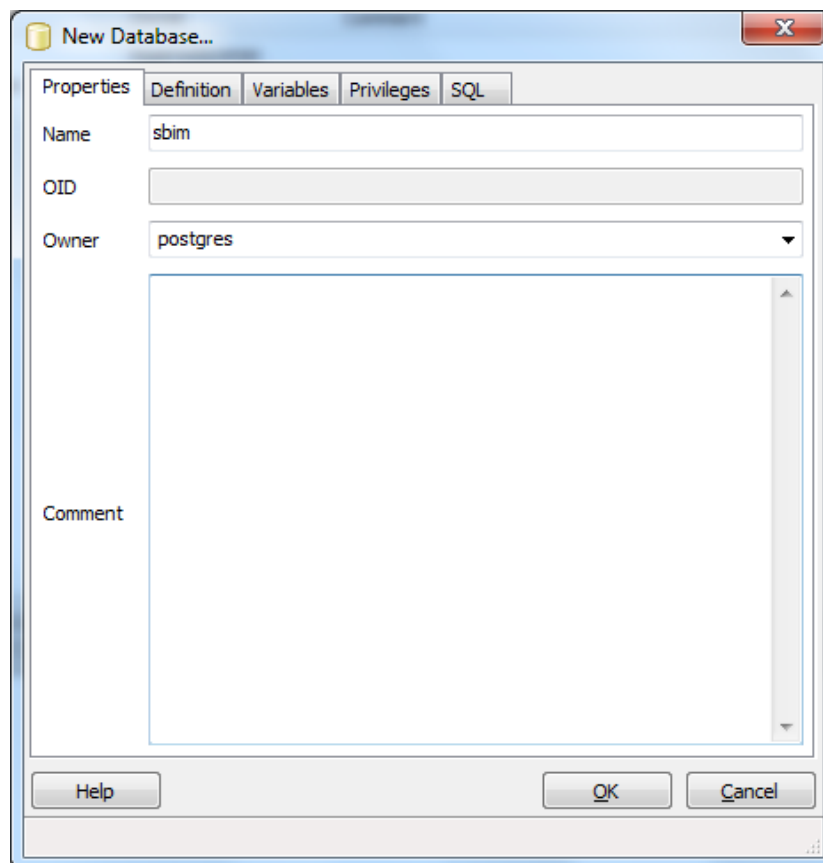


Figura 30 - Nova database

Após a criação da *database* é necessário realizar a importação do arquivo 'SBIM.backup', que está disponível no CD do projeto SBIM. A senha de acesso a base de dados é 'do6953'.

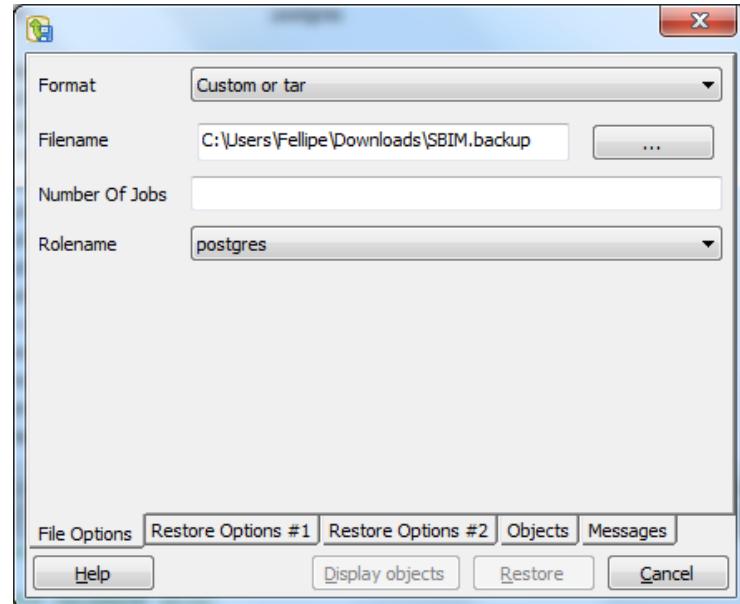


Figura 31 - Restauração de backup

Será criada a estrutura de tabelas a seguir, como mostra a figura 32.

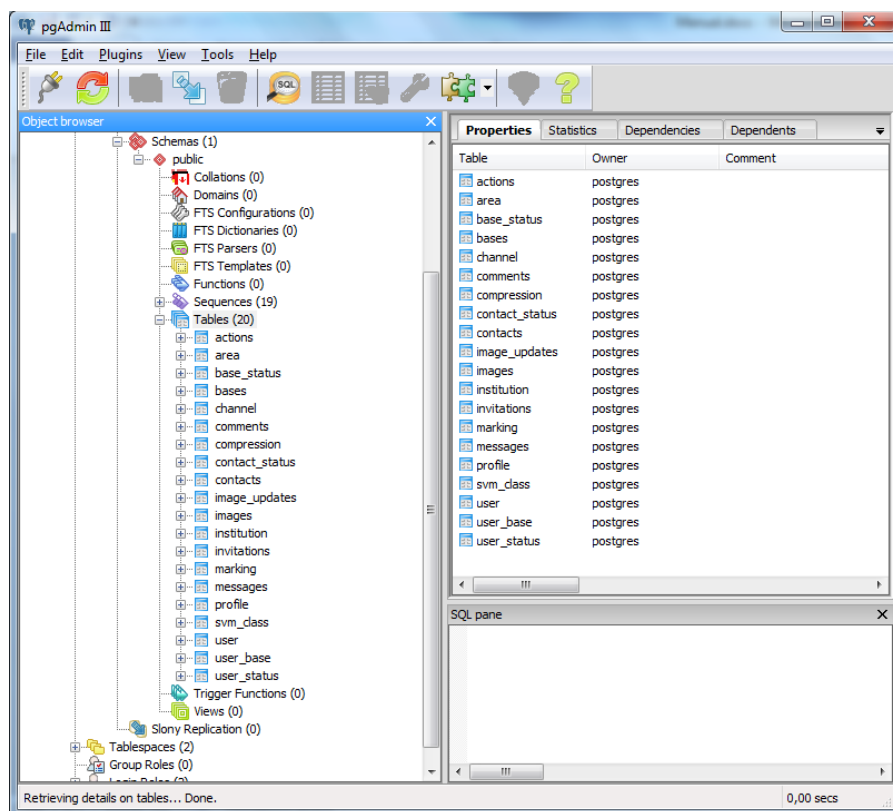


Figura 32 - Estrutura de tabelas

4.2.2 PASSO 2 – INSERIR O CÓDIGO FONTE NO APACHE TOMCAT

Com o Apache Tomcat desligado, insira o arquivo 'SBIM.war' na pasta C:\Program Files\Apache Software Foundation\Apache Tomcat 7.0.27\webapps (O local desta pasta pode ser diferente), certifique-se que a o Tomcat tem as permissões necessárias para a criação de pastas e inicie o servidor. O projeto será descompactado e instalado.

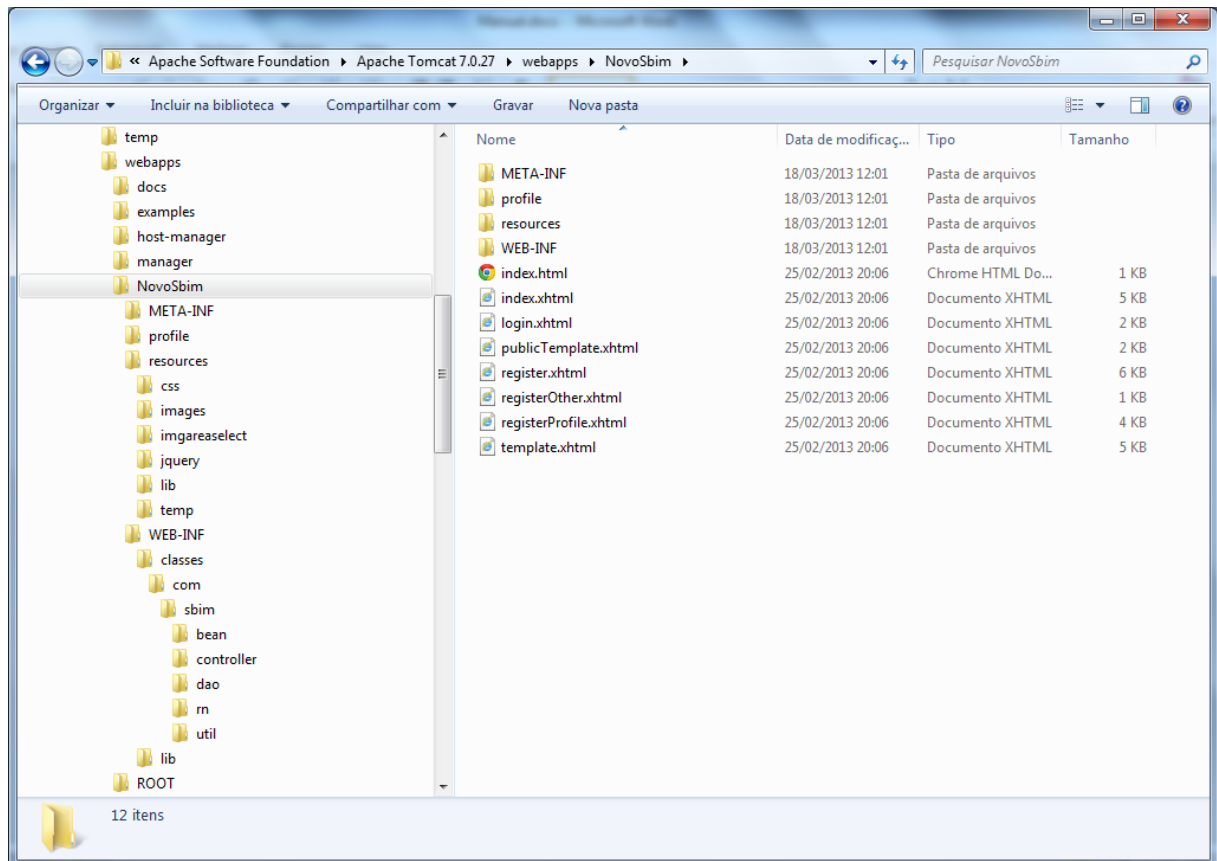


Figura 33 - Estrutura de pastas

4.2.3 PASSO 3 – ACESSAR A PÁGINA INICIAL DO SISTEMA SBIM

Após a conclusão dos passos anteriores, basta acessar a URL <http://localhost:8080/NovoSbim> para a realização do cadastro e a liberação de acesso ao SBIM.

5 CONCLUSÃO

Este trabalho foi a evolução do sistema SBIM já existente. As modificações ocorreram desde sua arquitetura, trazendo mais estabilidade e segurança ao sistema, até sua interface, deixando o sistema mais atrativo ao usuário.

A maior contribuição da nova concepção do sistema foi o desenvolvimento de uma rede social limpa, de fácil navegabilidade, com ferramentas de edição, marcação, classificação e recuperação de imagens utilizando conceitos de inteligência artificial. A interação entre os usuários através da Web contribui para diminuir as distancias entre pesquisadores de diferentes partes do país.

Entre os objetivos propostos para este projeto, listados brevemente a seguir, foram atingidos, com as seguintes contribuições:

- Desenvolvimento de uma rede social, auxiliando a interação entre os pesquisadores;
- Desenvolvimento de ferramentas para marcação e edição de imagens;
- Desenvolvimento do algoritmo SVM, permitindo a classificação e busca de imagens.

O *software* foi validado utilizando testes de integração, mostrando que todas as funcionalidades propostas funcionam corretamente. Também foi validada a acurácia do SVM, utilizando quatro bases de dados e verificando se o algoritmo retornava os resultados esperados. Na análise dos resultados obtidos foi observado que o SBIM teve um alto índice de acerto, com uma média de 97,50% de precisão. Com os resultados obtidos através dos testes chegamos a conclusão que o sistema desenvolvido atende aos objetivos propostos.

Entre as limitações do software desenvolvido podemos citar:

- Poucos filtros de imagem
- Problemas de desempenho quando o sistema possui bases com muitas imagens

- Busca de usuários limitada
- Falta de testes em produção com grande volume de usuários

Para estudos futuros, pretendemos:

- Utilizar CUDA (CUDA, 2013) para o aumento do desempenho do processamento de imagens.
- Aprimorar as funcionalidades de edição de imagens,
- Adição de mais filtros de imagens;
- Fazer uma versão *mobile* do SBIM;
- Fazer um administrador das bases do SBIM.

REFERÊNCIAS

ASTHA. Astah Community. Disponível em <<http://astah.net/editions/community>>

BARRETO, J. **Introdução às redes neurais artificiais**. 2002, p. 14 e 15.

BEAULIEU, A. **Aprendendo SQL**. Novatec, 2010.

BOOCH, G; RUMBAUGH J; JACOBSON, I. **UML Guia do Usuário**. Campus, 2000.

CODE. Google Code. Disponível em <<http://code.google.com>>

CONDUTA, B; MAGRIN, D. **Aprendizagem de Máquina**. 2010.

CUDA. Disponível em <http://www.nvidia.com/object/cuda_home_new.html>

DROPBOX . Disponível em <<http://www.dropbox.com>>

EAKINS, J; GRAHAM, M. **Content-based Image Retrieval**. University of New Castle, 2002.

FLANAGAN D. **Javascript - O Guia Definitivo**. Bookman, 2004.

GONZALEZ, R. C.; WOODS, R. E. **Processamento de Imagens Digitais**. Edgard Blucher, 2000.

GUEDES, G T A. **UML Uma Abordagem Prática**. Novatec, 2004.

HIBERNATE. JBoss Hibernate. Disponível em <<http://www.hibernate.org/>>

HORSTMANN, C. S.; CORNELL, G. **Core Java 2 Volume I - Fundamentos**. Makron Books, 2001.

HSU , C; CHIH, C; CHIH, L. **A Practical Guide to Support Vector Classification**. 2010, p.1 e p.1

KATO, T. **Database architecture for content-based image retrieval**. 1992.

LIBREOFFICE. Disponível em <<http://pt-br.libreoffice.org/>>

LIBSVM. Disponível em <<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>>

LORENA, A; CARVALHO, A. **Uma introdução às Support Vector Machines**. 2007, p53.

MARTINS, J C C. **Gerenciando Projetos de Desenvolvimento de Software** com PMI, RUP e UML. Brasport. 2004.

MATOS, F. **Reconhecimento de faces utilizando a transformada cosseno discreta**. UFP, 2008.

POSTGRESQL. Disponível em <<http://www.postgresql.org/>>

PRESSMAN, R. S. **Engenharia de Software**. Makron, 1995.

RUSSEL, S; NORVING, P. **Inteligência Artificial**. 2004.

SAVOLA, T. **Usando HTML**. Campus, 1997.

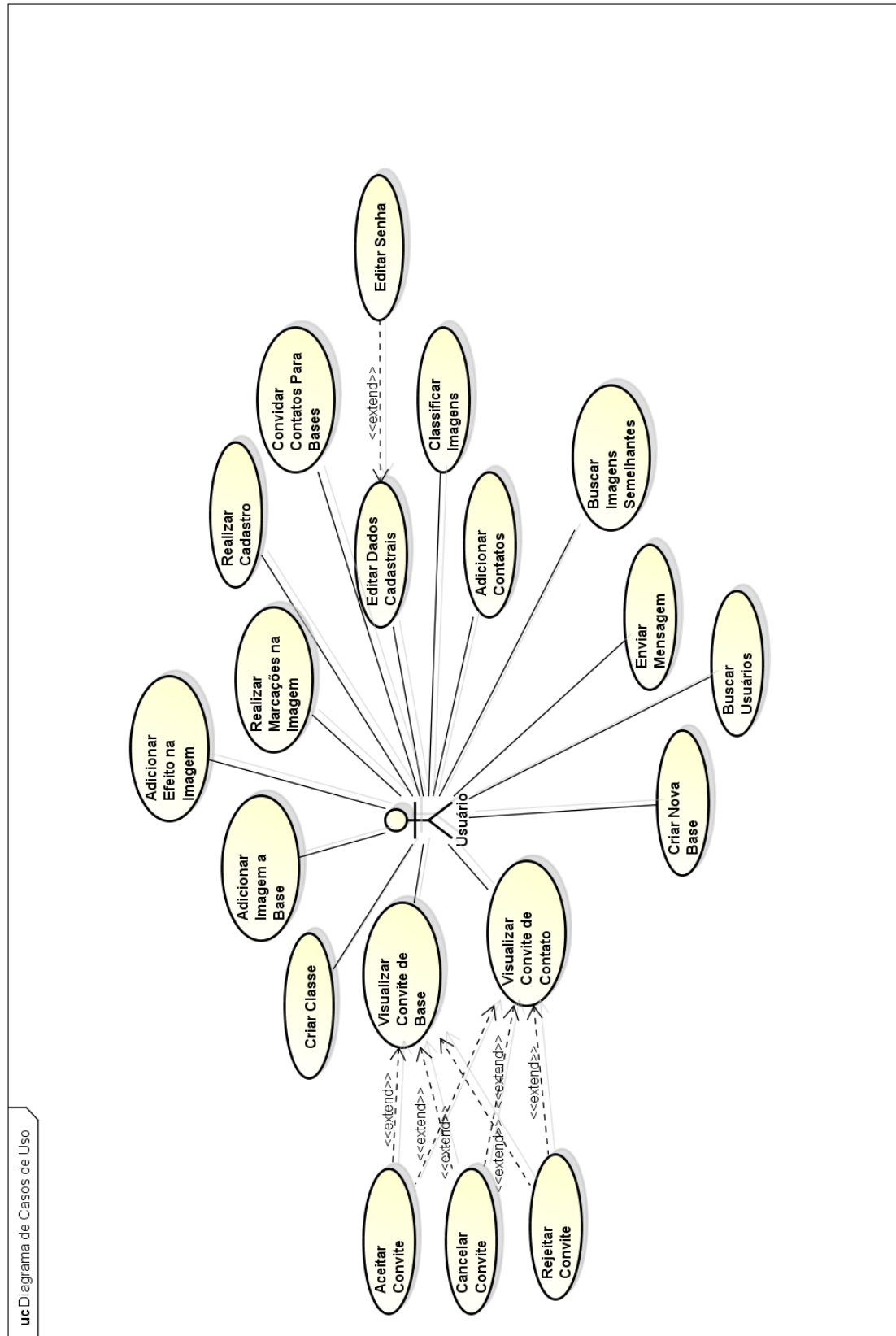
SMOLA, J; SCHÖLKOPF, B. **Learning with Kernels**. Cambridge: The MIT Press, 2002, p.14.

SUBVERSION. Apache Subversion. Disponível em <<http://subversion.apache.org/>>

TOMCAT. Apache Tomcat. Disponível em <<http://tomcat.apache.org/>>

WORKBENCH. MySQLWorkbench. Disponível em <<http://www.mysql.com/downloads/workbench/>>

APÊNCIDE A - DIAGRAMA DE CASO DE USO



Use Case	Realizar Cadastro
Pré-condições	Pessoa com interesse de ingressar no sistema, porém, sem cadastro.
Pós-condições	Usuário cadastrado no sistema, pronto para acessá-lo.
Sumário	Indica que a pessoa pode criar um novo cadastro no sistema.
Ator envolvido	Usuário.
Descrição	a) Os dados são preenchidos; b) Clica em “Register”; c) Os dados são verificados; (E1) d) O usuário é cadastrado e abre a tela inicial do SBIM.
Fluxo de exceção	E1) Campos obrigatórios não são preenchidos e/ou validados.

Use Case	Editar Dados de Usuário
Pré-condições	Usuário logado e na tela de edição de dados.
Pós-condições	Dados do usuário modificados.
Sumário	Indica que o usuário deseja alterar os seus dados.
Ator envolvido	Usuário.
Descrição	a) Usuário modifica os dados desejados; b) Clica no botão “Save Edit”; c) Dados são verificados; (E1) d) Dados do usuário são editados.
Fluxo de exceção	E1) Campos obrigatórios não são preenchidos e/ou validados.

Use Case	Enviar Mensagens
Pré-condições	Usuário logado e na tela de mensagens.
Pós-condições	Mensagem enviada para outro usuário.
Sumário	Indica que o usuário deseja enviar mensagens para outros usuários.
Ator envolvido	Usuário.
Descrição	a) O usuário busca/seleciona o usuário com quem quer conversar; b) Digita a mensagem desejada; c) O usuário clica no botão “Send Message”; d) O sistema verifica as regras de negócio; (E1) e) A mensagem é enviada.
Fluxo de exceção	E1) A descrição da mensagem é vazia.

Use Case	Buscar Usuários
Pré-condições	Usuário logado no sistema.
Pós-condições	Visualizar a página de outro usuário.
Sumário	Indica que o usuário deseja visualizar o perfil de outro usuário.
Ator envolvido	Usuário.
Descrição	a) O usuário digita o nome de quem ele está procurando; b) O sistema verifica se existe algum usuário que tem o nome pesquisado; (E1) c) É exibida a lista de usuários encontrados; d) O usuário clica no usuário desejado.
Fluxo de exceção	E1) O nome digitado não é encontrado.

Use Case	Criar Nova Base
Pré-condições	Usuário logado e na página de bases.
Pós-condições	Uma nova base é criada.
Sumário	Indica que o usuário deseja criar uma nova base de dados.
Ator envolvido	Usuário.
Descrição	a) O usuário clica no link “New Base”; b) São digitados os dados da base; c) O usuário clica no botão “Create Base”; d) O sistema faz a validação; (E1) e) A base é criada.
Fluxo de exceção	E1) O campo nome da base não foi preenchidos.

Use Case	Adicionar Imagem a Base
Pré-condições	Usuário logado, pertencendo a uma base e na página de gerenciamento de bases.
Pós-condições	Uma nova imagem é adicionada a base correspondente.
Sumário	Indica que o usuário deseja adicionar uma nova imagem a uma base de dados que ele pertence.
Ator envolvido	Usuário.
Descrição	a) O usuário clica no botão “+Add”; b) A imagem desejada é selecionada; (E1) c) O usuário clica no botão “Upload”; d) A validação do sistema é executada; (E2)

	e) A imagem é adicionada e a pagina re-renderizada.
Fluxo de exceção	E1) O arquivo selecionado não é uma imagem, ou então não é um formato de imagem suportado. E2) O sistema não tem permissão de escrita na pasta necessária para a escrita do arquivo.

Use Case	Marcar Imagem
Pré-condições	Usuário logado, pertencendo a uma base com uma imagem disponível e na página de edição de imagem.
Pós-condições	Uma nova marcação é feita na imagem.
Sumário	Indica que o usuário deseja fazer uma nova marcação na imagem.
Ator envolvido	Usuário.
Descrição	a) O usuário seleciona a área desejada para a marcação; b) O usuário digita a descrição da marcação e tecla "Enter"; c) A marcação é adicionada pelo sistema; d) O usuário clica no link "Save Edit"; e) O sistema salva a marcação.
Fluxo de exceção	

Use Case	Adicionar Contatos
Pré-condições	Usuário logado no sistema e na página de perfil de outro usuário.
Pós-condições	Um convite de contato é enviado para o outro usuário.
Sumário	Indica que o usuário deseja convidar um outro usuário para ser seu contato.
Ator envolvido	Usuário.
Descrição	a) O usuário clica no botão "Add to contacts"; b) O sistema envia um pedido de contato para o outro usuário.
Fluxo de exceção	

Use Case	Convidar Contatos Para Bases
Pré-condições	Usuário logado no sistema, pertencendo a uma base e na pagina de perfil de

	outro usuário.
Pós-condições	Um convite para participar da base selecionada é enviada a outro contato.
Sumário	Indica que o usuário deseja convidar outro usuário para participar de uma base.
Ator envolvido	Usuário.
Descrição	a) O usuário clica no botão “Invite to a group”; b) O popup com as bases disponíveis é aberto; c) O usuário escolhe a base e clica no botão “Invite”; d) O sistema envia um pedido de participação da base para o outro usuário.
Fluxo de exceção	

Use Case	Classificar Imagens
Pré-condições	Usuário logado no sistema, com uma classe disponível, uma imagem não classificada e na página de classificação.
Pós-condições	As imagens selecionadas serão classificadas.
Sumário	Indica que o usuário deseja classificar imagens.
Ator envolvido	Usuário.
Descrição	a) O usuário seleciona as imagens desejada; b) É selecionada a classe; c) O sistema classifica a imagem de acordo com a classe selecionada.
Fluxo de exceção	

Use Case	Buscar Imagens Semelhantes
Pré-condições	Usuário logado no sistema e na página de busca de imagens.
Pós-condições	É mostrada a lista de imagens semelhantes.
Sumário	Indica que o usuário deseja buscar as imagens semelhantes a aquela que foi feita o upload.
Ator envolvido	Usuário.
Descrição	a) O usuário clica no botão “+Add”; b) A imagem desejada é selecionada; (E1)

	c) O usuário clica no botão "Upload"; d) O sistema salva a imagem em uma pasta temporária (E2); e) É feita a chamada ao SVM, que verifica as imagens semelhantes a aquela que foi feito o upload; f) As imagens são exibidas.
Fluxo de exceção	E1) O arquivo selecionado não é uma imagem, ou então não é um formato de imagem suportado. E2) O sistema não tem permissão de escrita na pasta necessária para a escrita do arquivo.

Use Case	Adicionar Efeito na Imagem
Pré-condições	Usuário logado, pertencendo a uma base com uma imagem disponível e na página de edição de imagem.
Pós-condições	O efeito é aplicado na imagem.
Sumário	Indica que o usuário deseja aplicar um efeito na imagem.
Ator envolvido	Usuário.
Descrição	a) O usuário clica no link referente ao efeito desejado; b) O efeito é aplicado.
Fluxo de exceção	

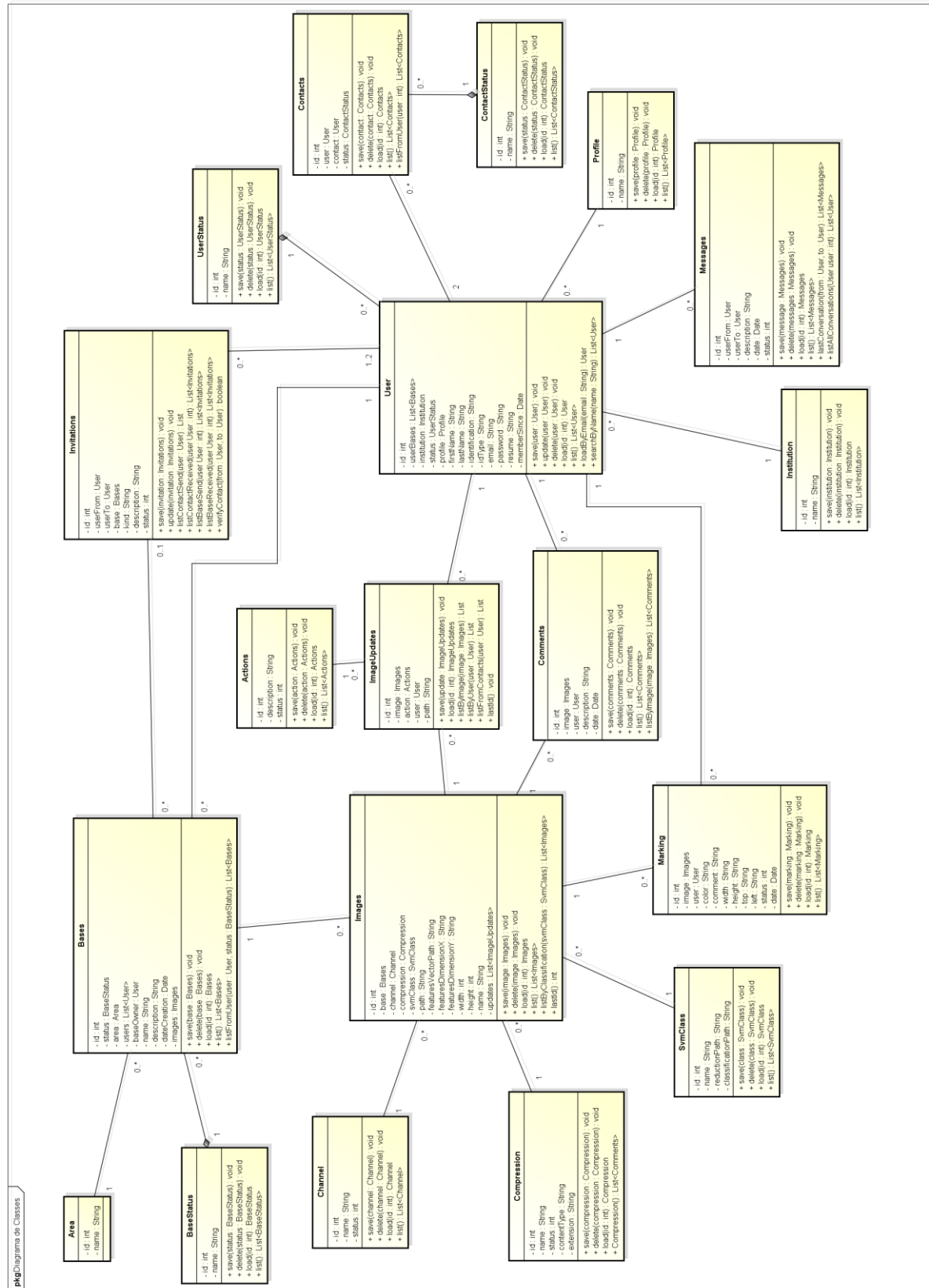
Use Case	Visualizar Convite de Base
Pré-condições	Usuário logado, com um convite para base pendente.
Pós-condições	O convite visualizado.
Sumário	Indica que o usuário deseja visualizar seus convites referente a bases.
Ator envolvido	Usuário.
Descrição	a) O usuário abre a página de convites b) O convite é visualizado.
Fluxo de exceção	

Use Case	Visualizar Convite de Contato
Pré-condições	Usuário logado, com um convite para contato pendente.

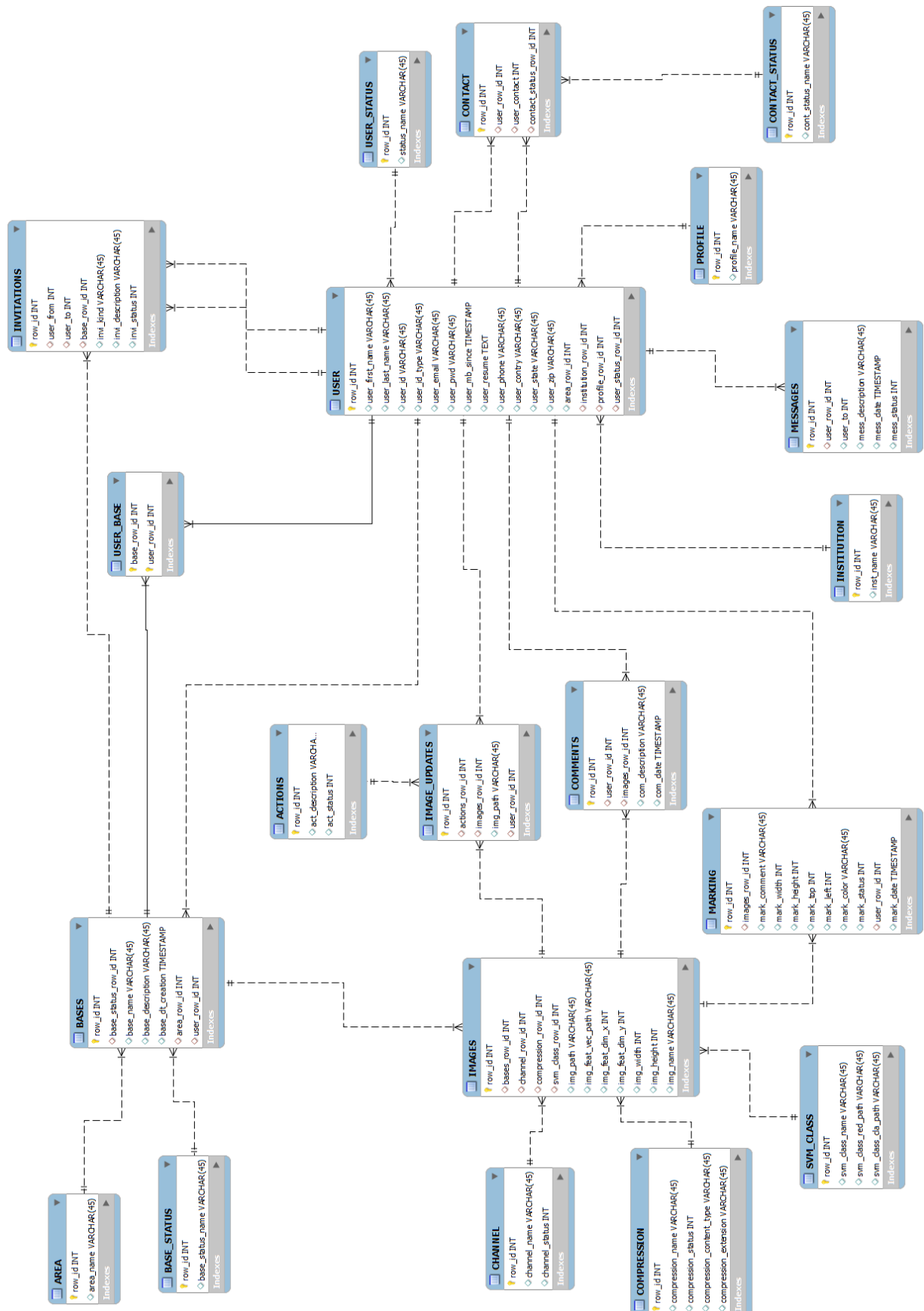
Pós-condições	O convite visualizado.
Sumário	Indica que o usuário deseja visualizar seus convites referente a contatos.
Ator envolvido	Usuário.
Descrição	a) O usuário abre a página de convites b) O convite é visualizado.
Fluxo de exceção	

Use Case	Criar Nova Classe
Pré-condições	Usuário logado e na página de classes.
Pós-condições	Uma nova classe é criada.
Sumário	Indica que o usuário deseja criar uma nova classe.
Ator envolvido	Usuário.
Descrição	a) O usuário clica no link “New Class”; b) São digitados os dados da classe; c) O usuário clica no botão “Create Class”; d) O sistema verifica as regras de negócio; (E1) e) A classe é criada.
Fluxo de exceção	E1) Os campos obrigatórios não foram preenchidos.

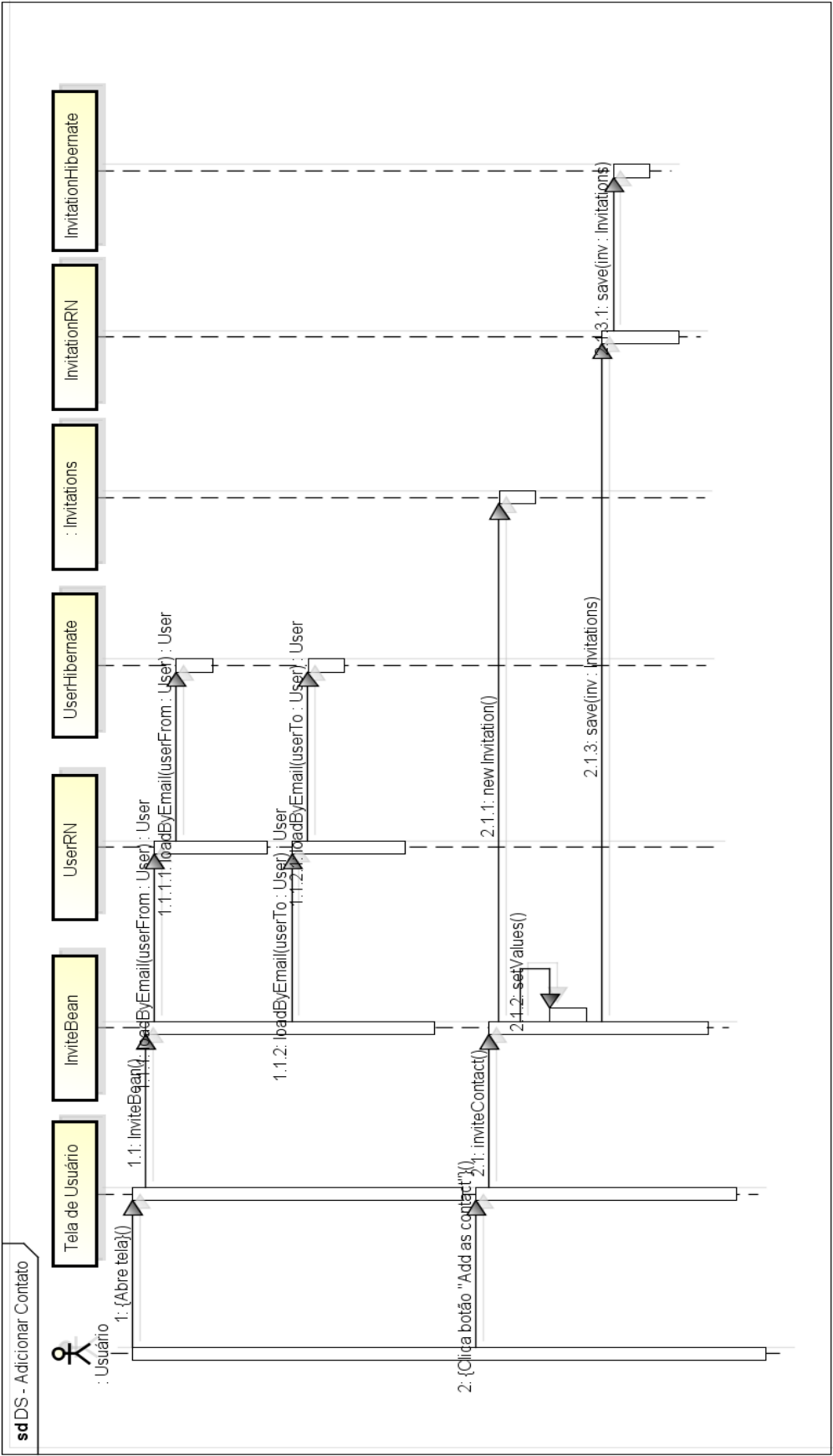
APÊNCIDE B - DIAGRAMA DE CLASSE

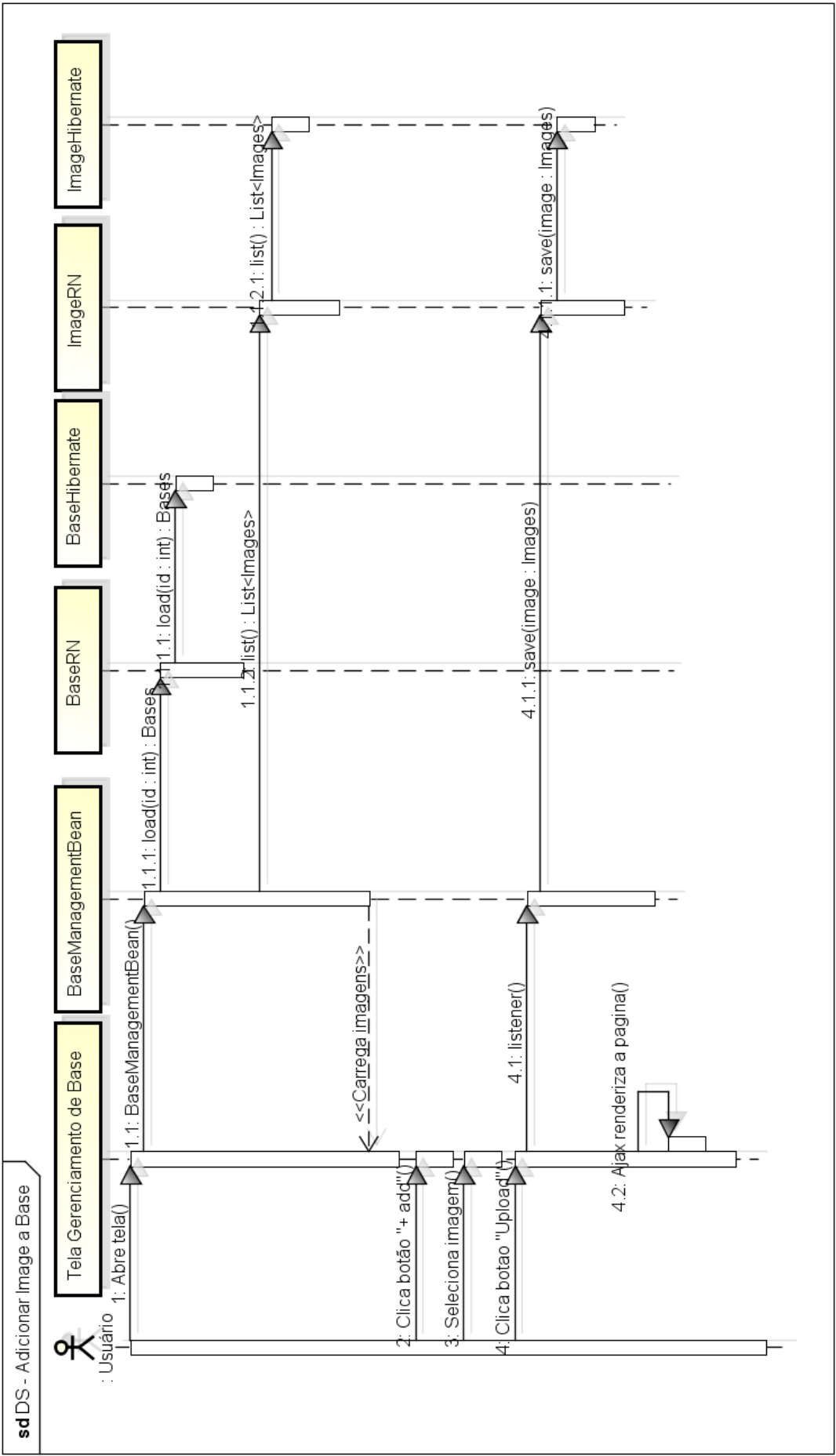


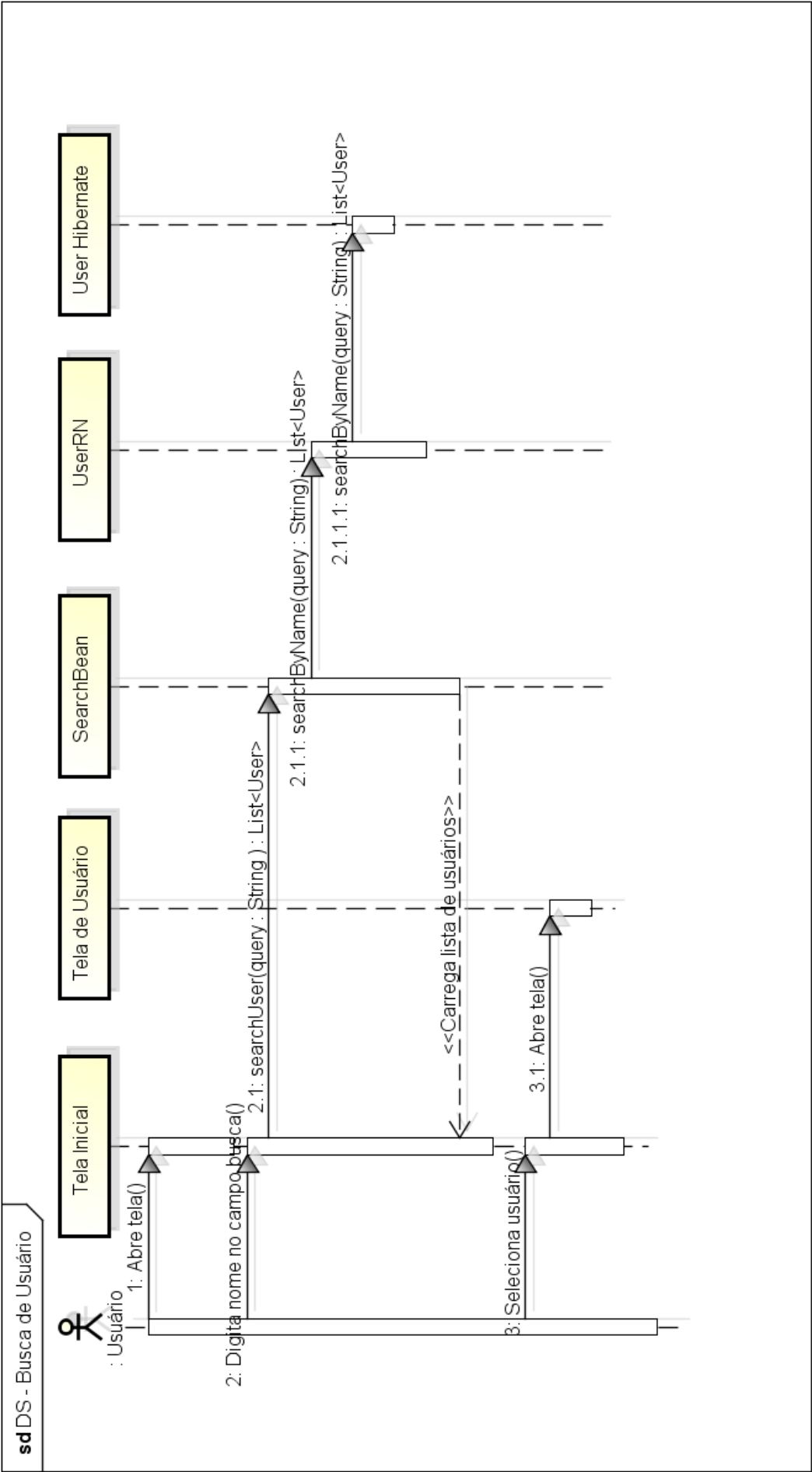
APÊNCIDE C - DIAGRAMA ENTIDADE RELACIONAMENTO

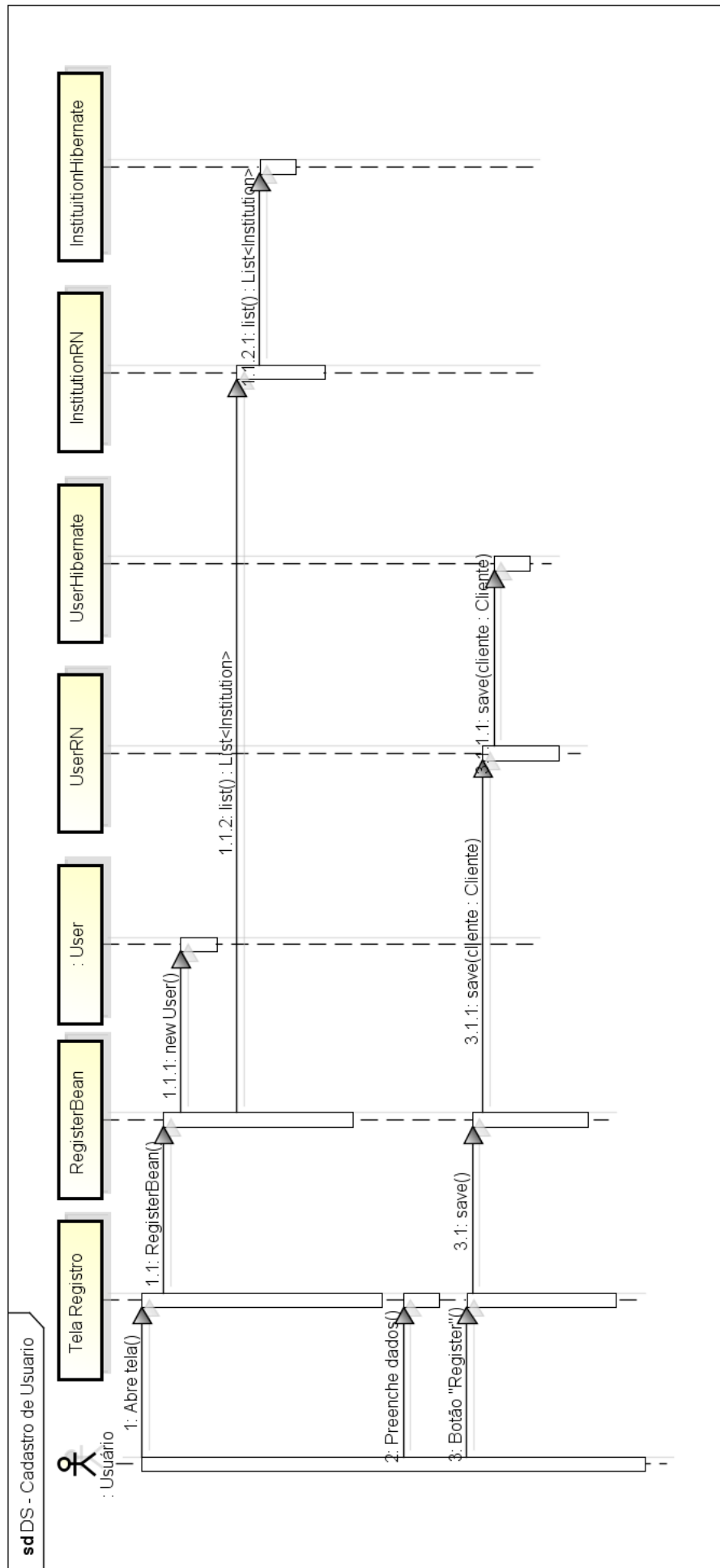


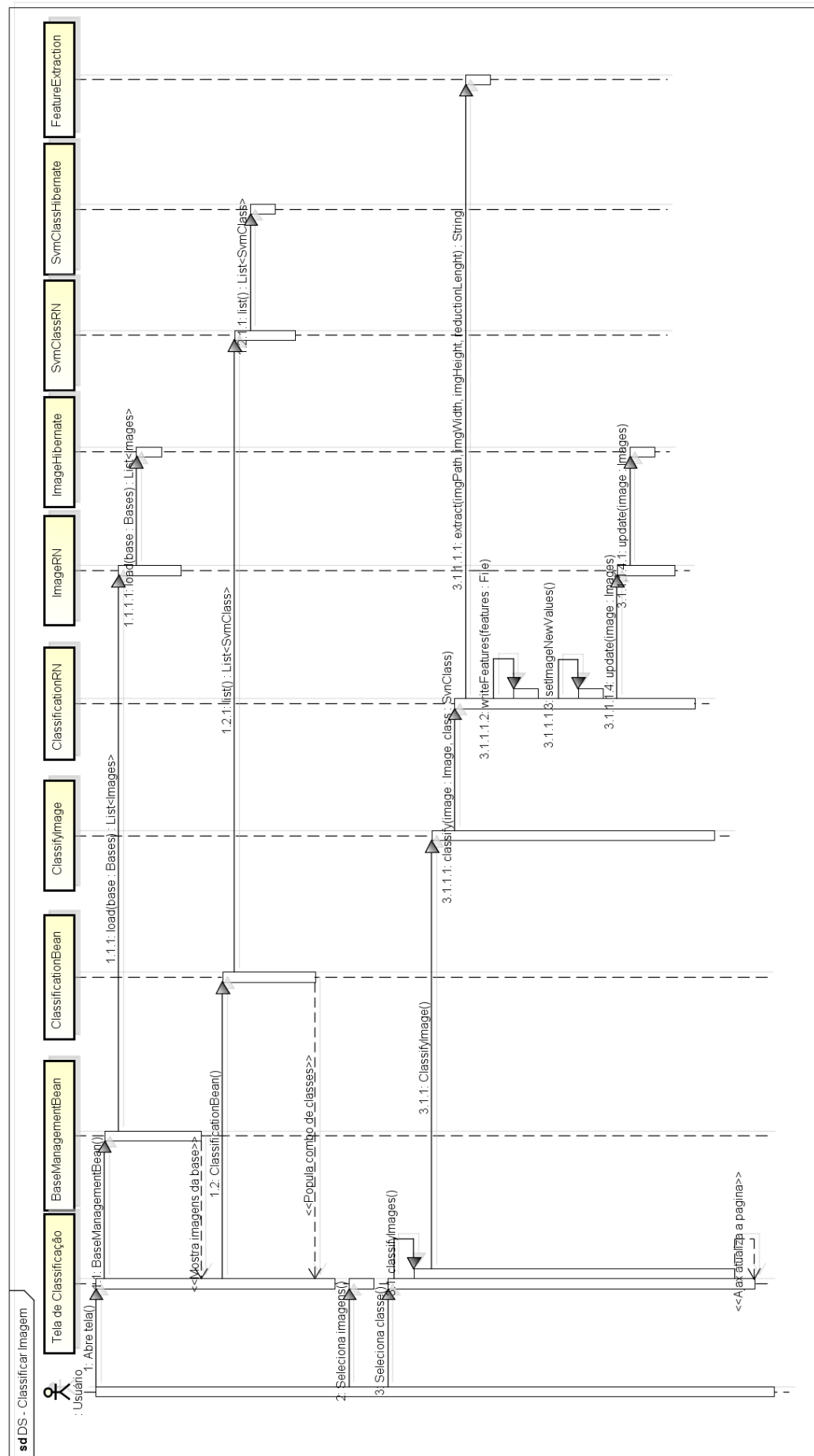
APÊNCIDE D - DIAGRAMA DE SEQUENCIA

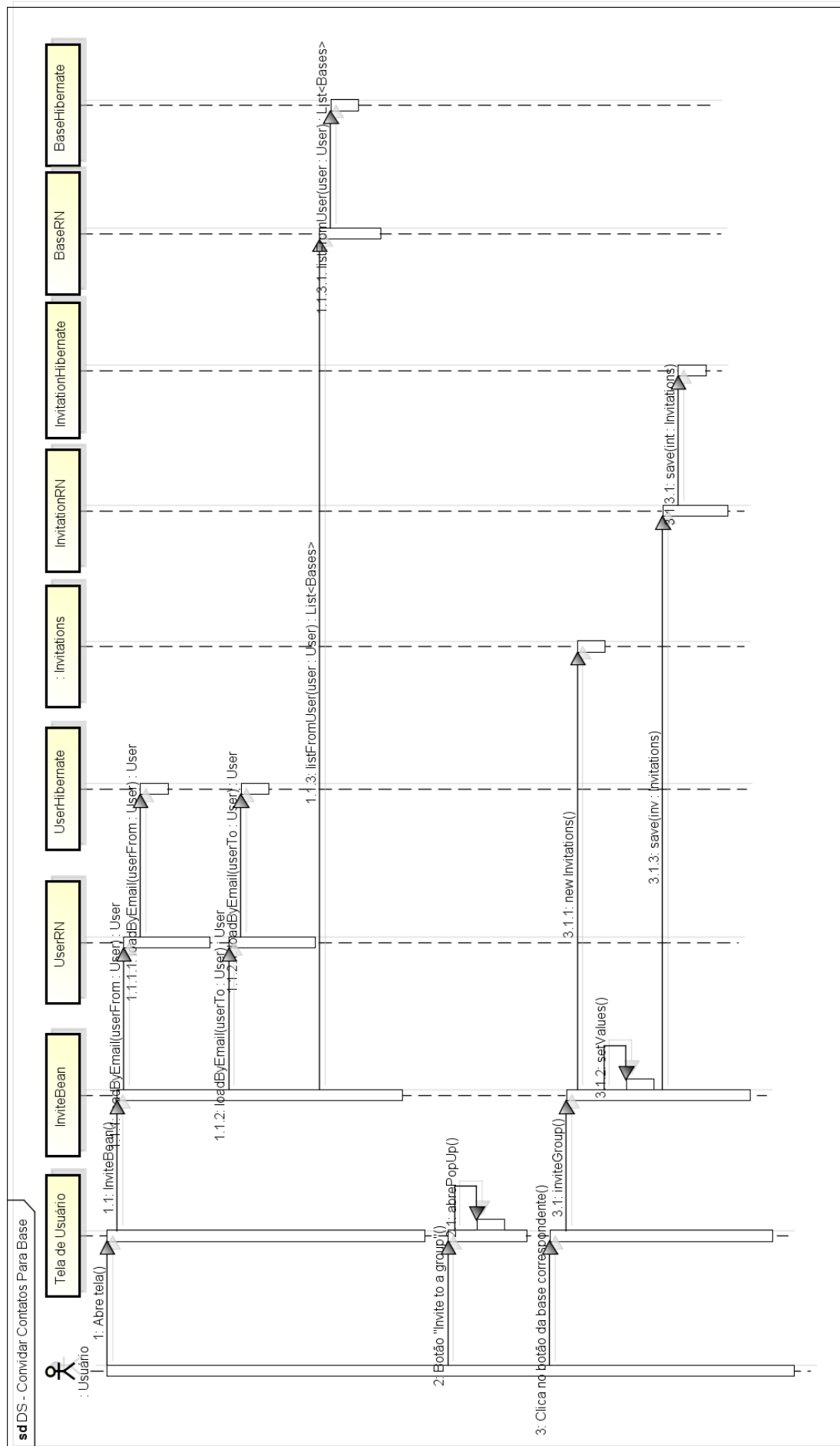


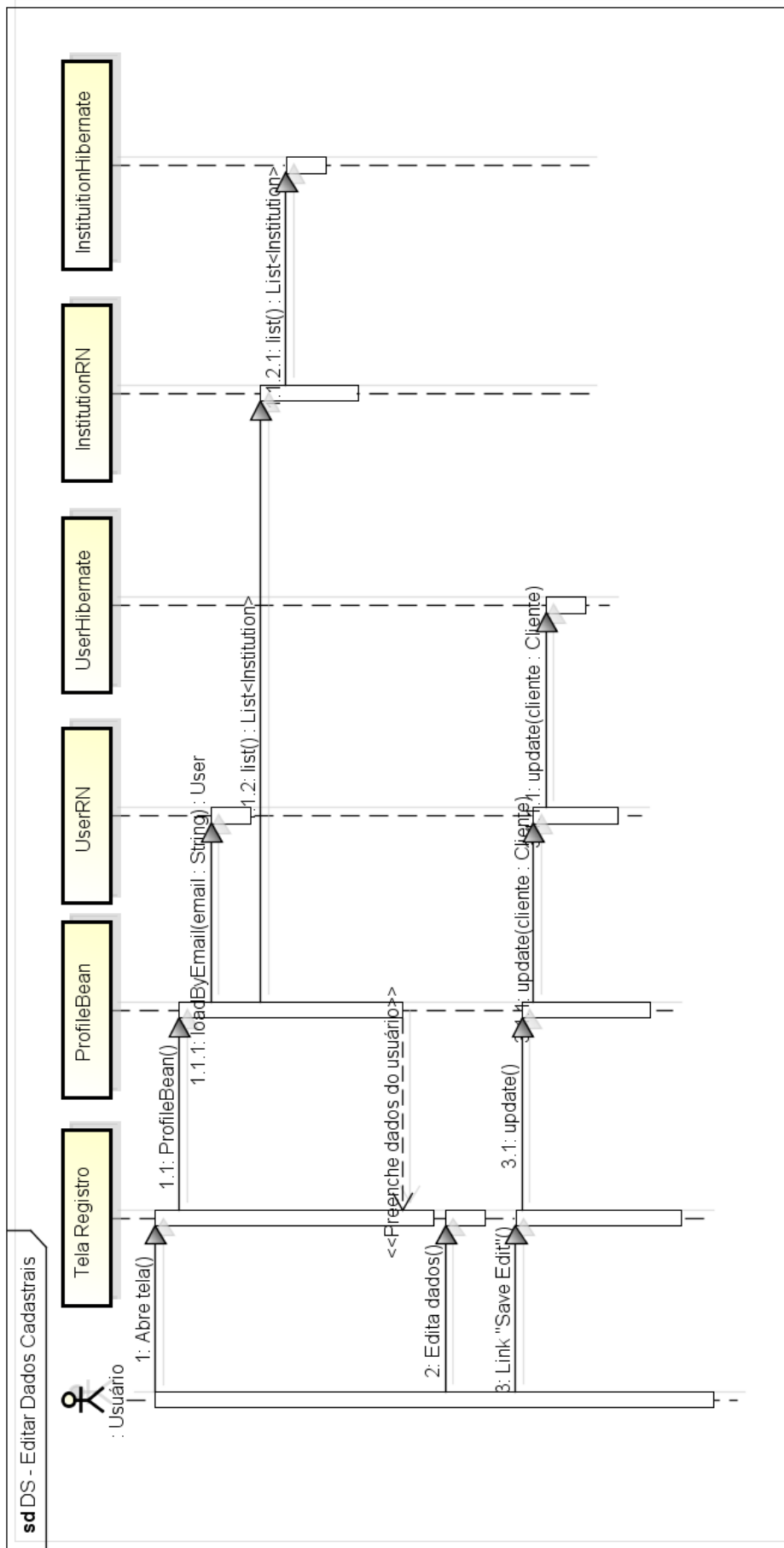


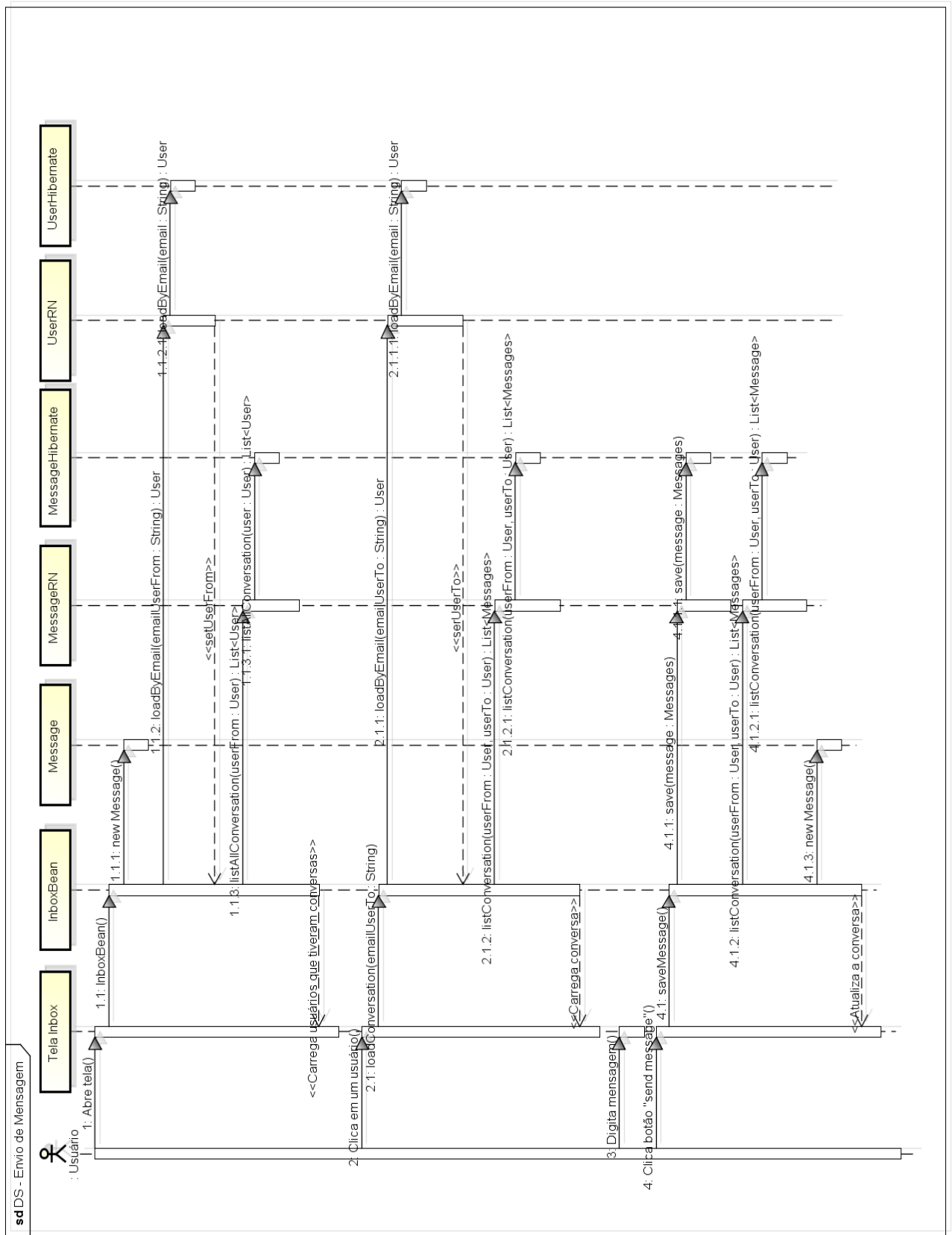


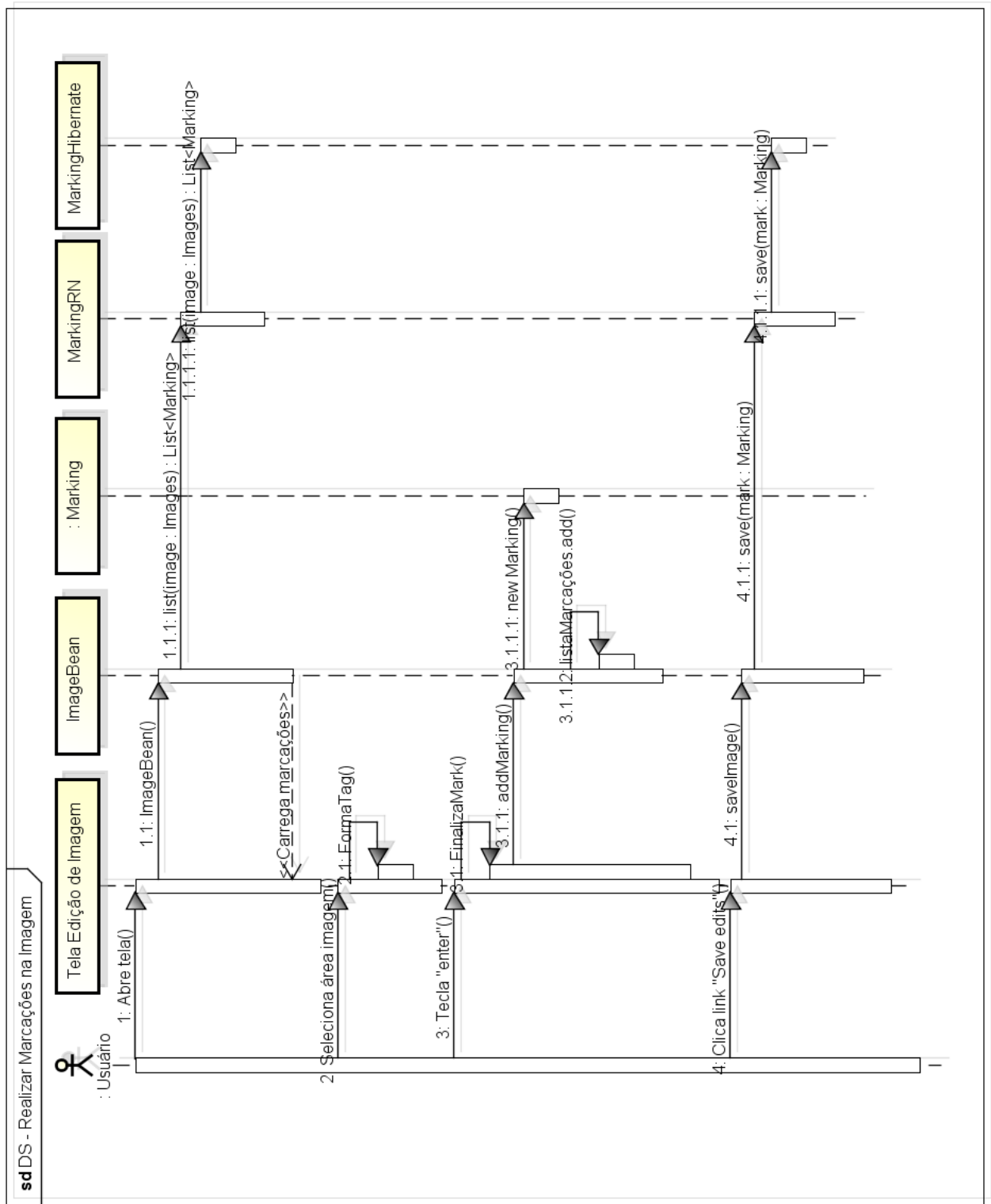


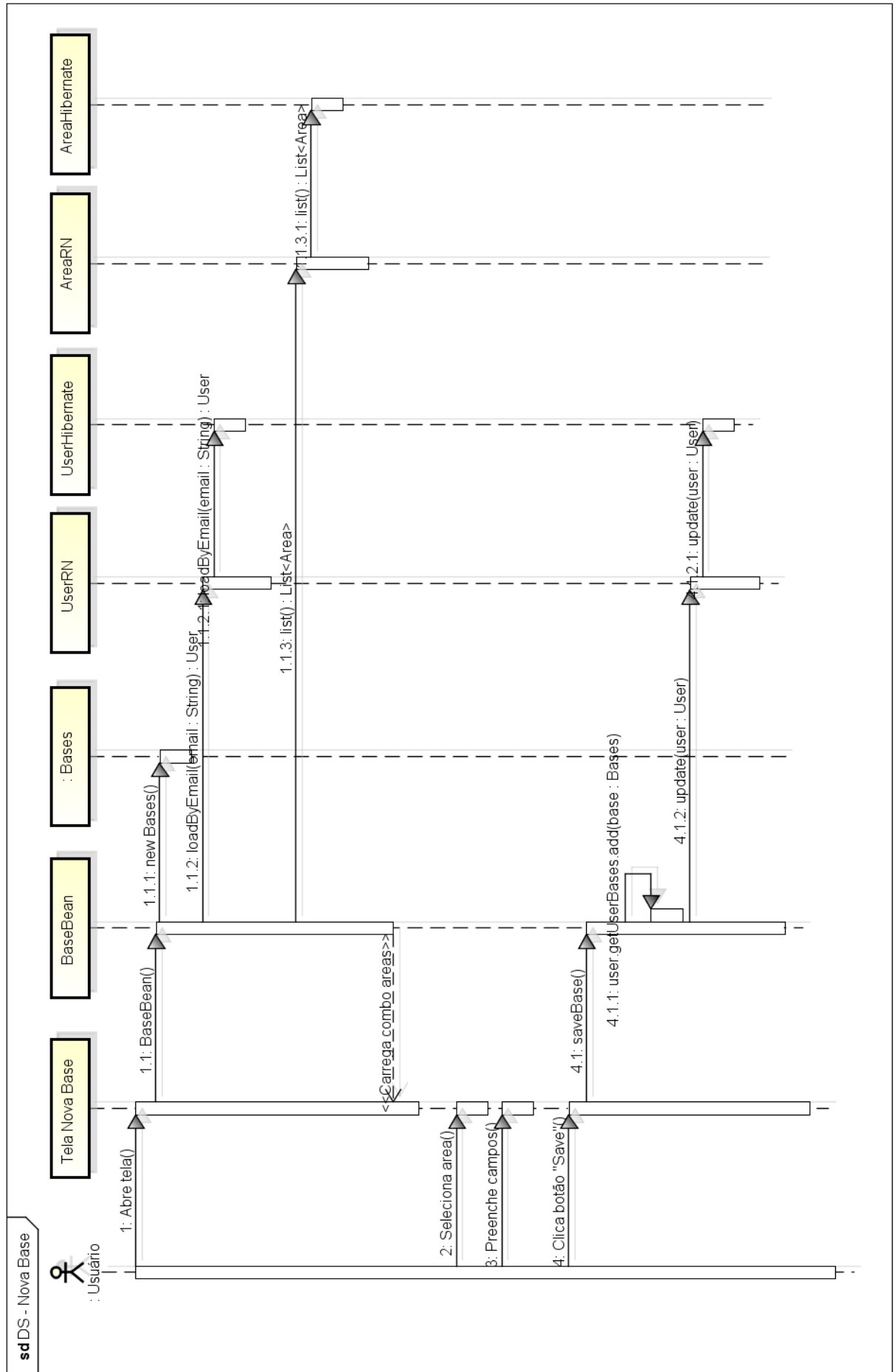


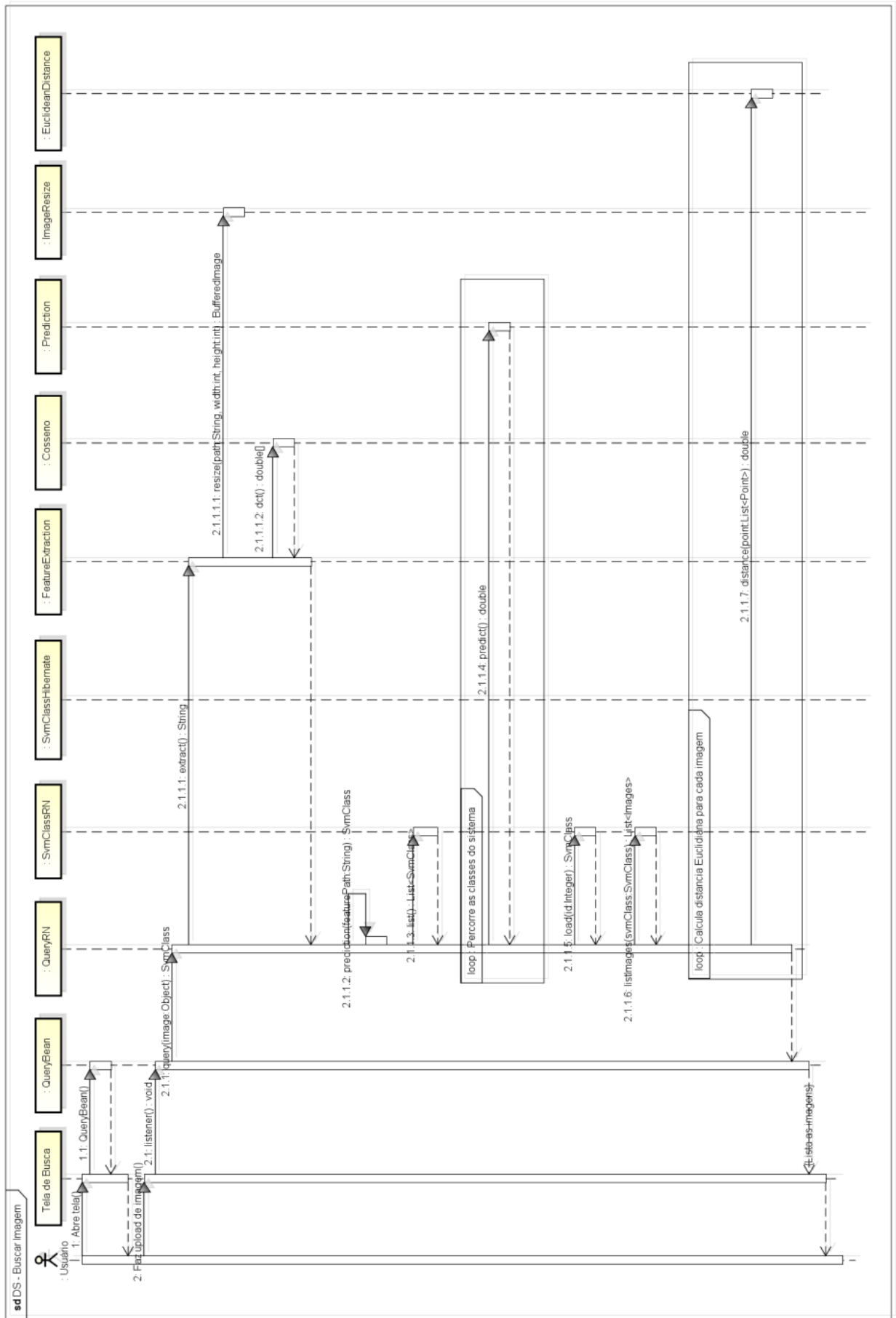


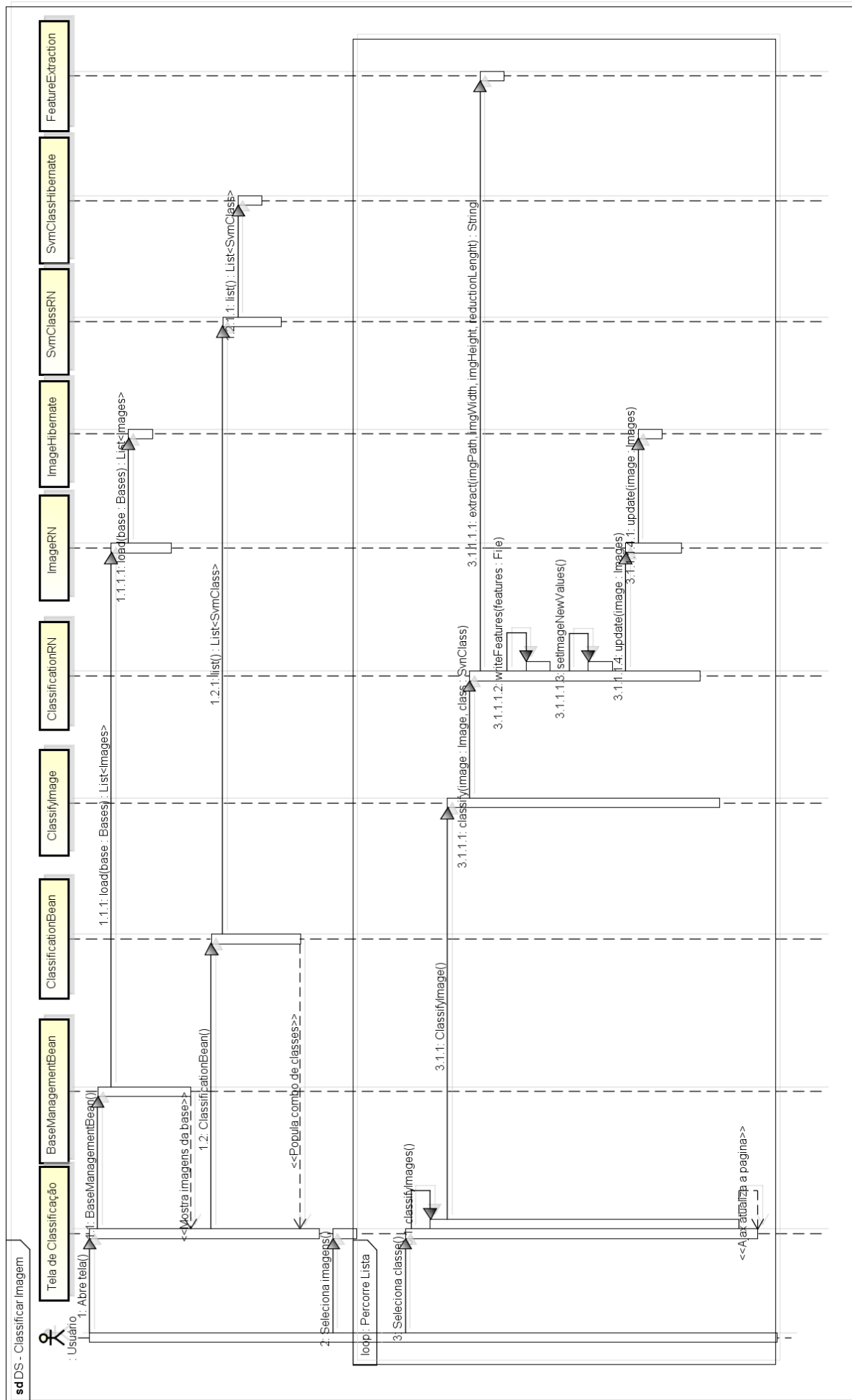












APÊNCIDE E - DICIONÁRIO DE DADOS

Tabela 'actions'		
Atributo	Domínio	Descrição
row_id	INT	Código da ação (PK).
act_description	VARCHAR	Descrição da ação.
act_status	INT	Status da ação.

Tabela 'area'		
Atributo	Domínio	Descrição
row_id	INT	Código da área (PK).
area_name	VARCHAR	Nome da área.

Tabela 'base_status'		
Atributo	Domínio	Descrição
row_id	INT	Código do status da base (PK).
base_status_name	VARCHAR	Nome do status.

Tabela 'bases'		
Atributo	Domínio	Descrição
row_id	INT	Código da base (PK).
base_status_row_id	INT	Status da base (FK).
base_name	VARCHAR	Nome da base.
base_description	VARCHAR	Descrição da base.
base_dt_creation	DATE	Data de criação da base.
area_row_id	INT	Código da área da base. (FK)
user_row_id	INT	Código do usuário que criou a base (FK).

Tabela 'channel'		
Atributo	Domínio	Descrição
row_id	INT	Código do canal (PK).
channel_name	VARCHAR	Nome do canal.
channel_status	INT	Status do canal.

Tabela 'comments'		
Atributo	Domínio	Descrição
row_id	INT	Código do comentário (PK).
user_row_id	INT	Código do usuário que fez o comentário (FK).
images_row_id	INT	Código da imagem a que pertence o comentário (FK).
com_description	VARCHAR	O que foi escrito no comentário.
com_date	DATE	Data em que foi feito o comentário.

Tabela 'compression'		
Atributo	Domínio	Descrição
row_id	INT	Código da compressão (PK).
compression_name	VARCHAR	Nome da compressão.
compression_status	INT	Status da compressão.
compression_content_type	VARCHAR	Content type da compressão.
compression_extension	VARCHAR	Extensão da compressão.

Tabela 'contact_status'		
Atributo	Domínio	Descrição
row_id	INT	Código do status do contato (PK).
cont_status_name	VARCHAR	Nome do status do contato.

Tabela 'contact'		
[Mostra as 'amizades' entre os usuários]		
Atributo	Domínio	Descrição
row_id	INT	Código do contato (PK).
contact_status_row_id	INT	Código do status do contato (FK).
user_row_id	INT	Código do usuário que fez a solicitação de contato (FK).
user_contact	INT	Código do usuário que recebeu a solicitação de contato (FK).

Tabela 'image_updates'		
Atributo	Domínio	Descrição
row_id	INT	Código do update (PK).
actions_row_id	INT	Código da ação que gerou o update (FK).
images_row_id	INT	Código da imagem que recebeu o update (FK).
user_row_id	INT	Código do usuário que fez o update (FK).

Tabela 'images'		
Atributo	Domínio	Descrição
row_id	INT	Código da imagem (PK).
bases_row_id	INT	Código da base a qual pertence a imagem (FK).
channel_row_id	INT	Código do canal da imagem (FK).
compression_row_id	INT	Código do formato de compressão da imagem (FK).
svm_class_row_id	INT	Código da classe SVM da imagem (FK).
img_path	INT	Local físico em que a imagem foi armazenada.
img_feat_vec_path	VARCHAR	Local físico das características da imagem.
img_feat_dim_x	INT	Tamanho em pixels do vértice X das características da

		imagem.
img_feat_dim_y	INT	Tamanho em pixels do vértice Y das características da imagem.
img_width	INT	Largura em pixels da imagem.
img_height	INT	Altura em pixels da imagem.

Tabela 'institution'		
Atributo	Domínio	Descrição
row_id	INT	Código da instituição. (PK)
inst_name	VARCHAR	Nome da instituição.

Tabela 'invitations'		
Atributo	Domínio	Descrição
row_id	INT	Código do convite. (PK)
user_from	INT	Código do usuário que enviou o convite. (FK)
user_to	INT	Código do usuário que recebeu o convite. (FK)
base_row_id	INT	Código da base em que se refere o convite. (FK)
invi_kind	INT	Tipo do convite.
invi_description	INT	Descrição do convite.
status	VARCHAR	Status do convite.

Tabela 'marking'		
Atributo	Domínio	Descrição
row_id	INT	Código da marcação. (PK)
images_row_id	INT	Código da imagem marcada. (FK)
mark_comment	VARCHAR	Descrição/nome da marcação.
mark_width	VARCHAR	Largura em pixels da marcação
mark_height	VARCHAR	Altura em pixels da marcação.
mark_top	VARCHAR	Ponto superior em pixels da marcação. (Ponto inicial)

mark_left	VARCHAR	Ponto mais a direita em pixels da marcação. (Ponto final)
mark_color	VARCHAR	Cor da marcação.
mark_status	INT	Status da marcação.
mark_date	DATE	Data em que foi feita a marcação.

Tabela 'messages'		
Atributo	Domínio	Descrição
row_id	INT	Código da mensagem. (PK)
user_row_id	INT	Código do usuário que enviou a mensagem. (FK)
user_to	INT	Código do usuário que recebeu a mensagem. (FK)
mess_description	VARCHAR	Descrição/conteúdo da mensagem.
mess_date	DATE	Data em que foi enviada a mensagem.
mess_status	INT	Status da mensagem.

Tabela 'profile'		
Atributo	Domínio	Descrição
row_id	INT	Código do perfil. (PK)
profile_name	VARCHAR	Nome do perfil.

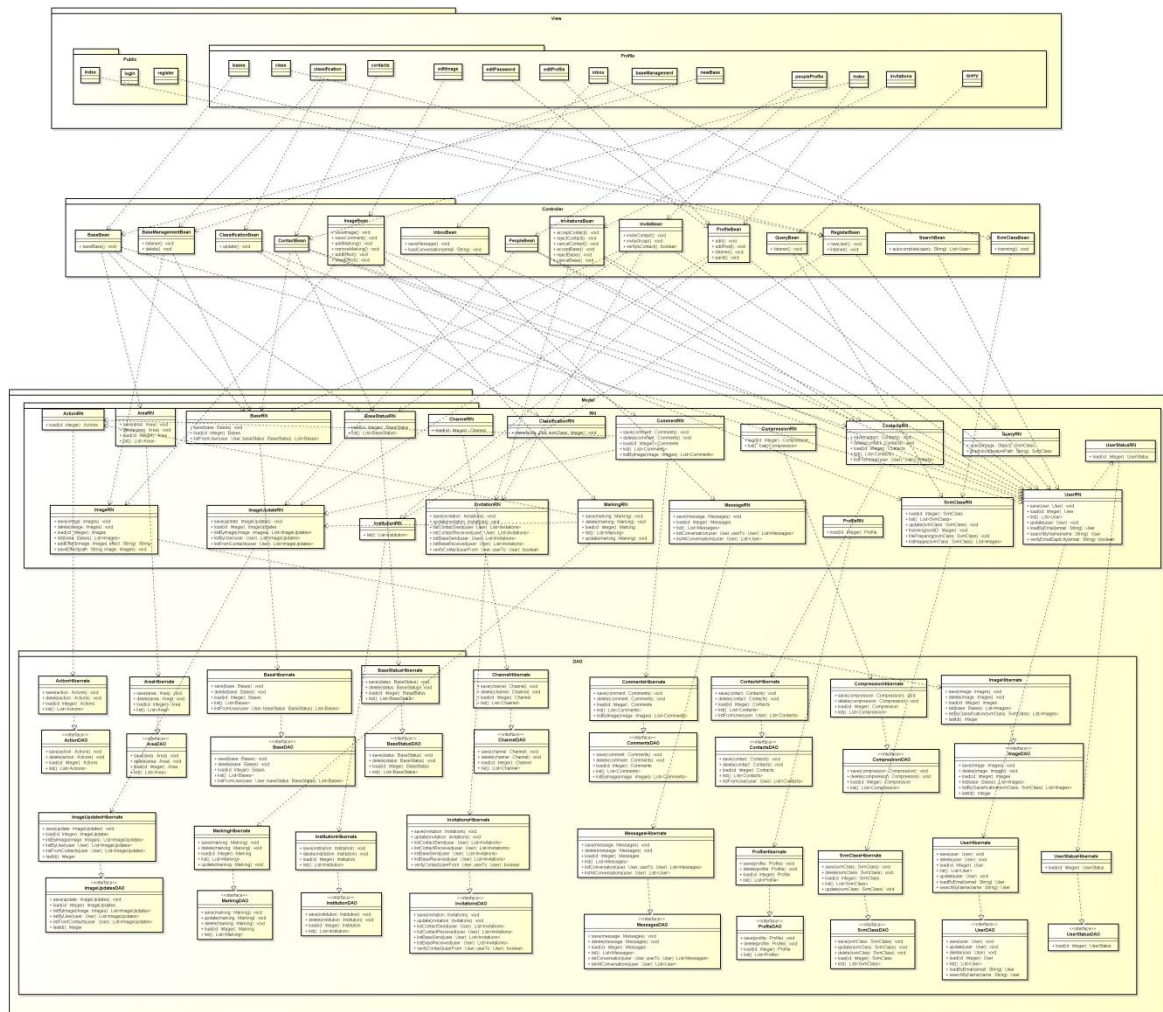
Tabela 'svm_class'		
Atributo	Domínio	Descrição
row_id	INT	Código da classe SVM. (PK)
svm_class_name	VARCHAR	Nome da classe SVM.
svm_class_red_path	VARCHAR	
svm_class_cla_path	VARCHAR	

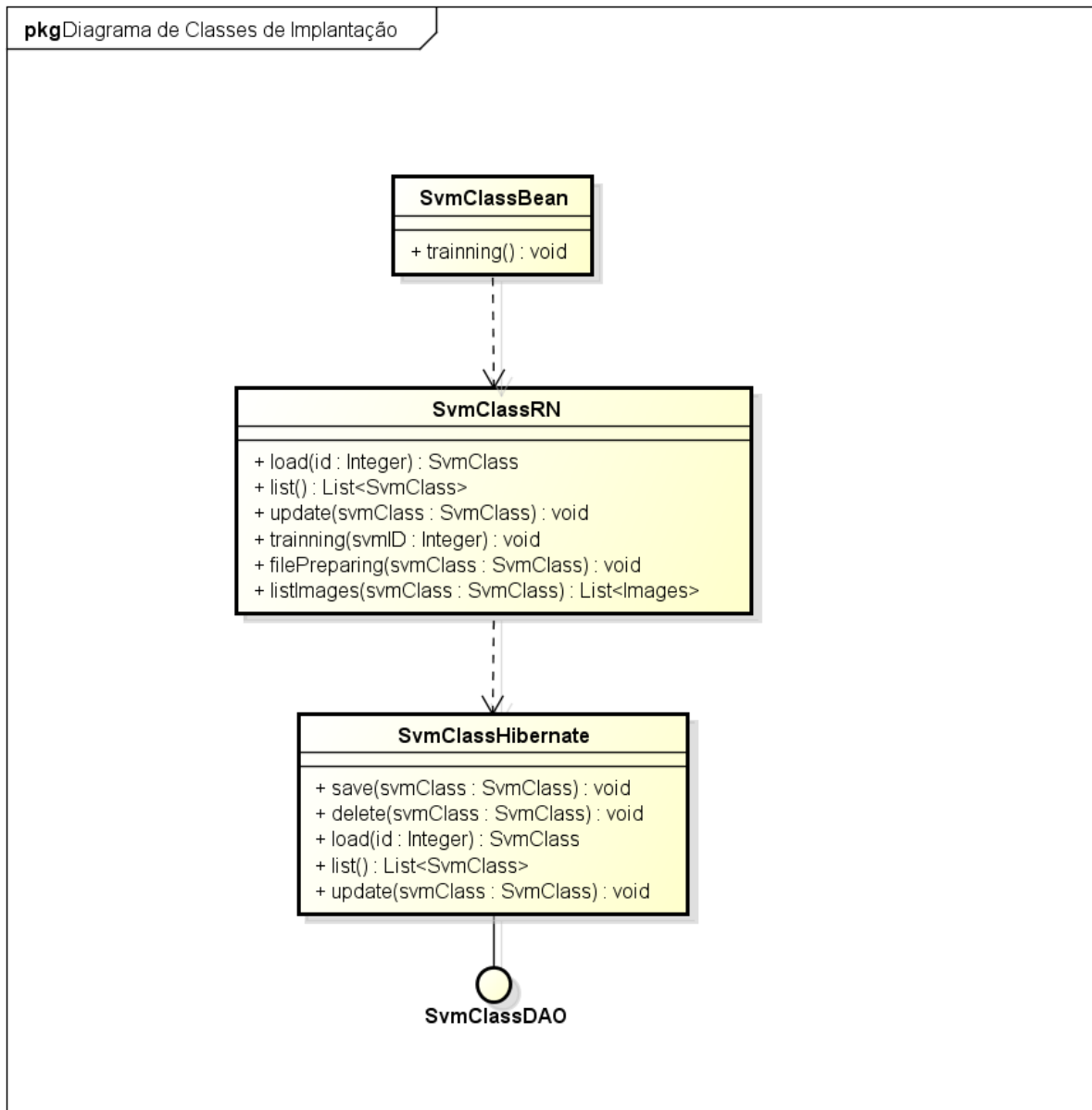
Tabela 'user'		
Atributo	Domínio	Descrição
row_id	INT	Código do usuário. (PK)
institution_row_id	INT	Código da instituição do usuário. (FK)
profile_row_id	INT	Código do perfil do usuário. (FK)
user_status_row_id	INT	Código do status do usuário. (FK)
user_first_name	INT	Primeiro nome do usuário.
user_last_name	INT	Último nome do usuário.
user_id	VARCHAR	Identificação do usuário.
user_id_type	VARCHAR	Tipo da identificação.
user_email	VARCHAR	Email do usuário.
user_pwd	VARCHAR	Senha do usuário.
user_resume	VARCHAR	Pequeno currículo do usuário.
user_a_of_interest	VARCHAR	Área de interesse do usuário
user_phone	VARCHAR	Telefone do usuário.
user_country	VARCHAR	País do usuário.
user_state	VARCHAR	Estado/provincia do usuário.
user_city	VARCHAR	Cidade do usuário.
user_zip_code	VARCHAR	ZIP code/ CEP do usuário.
user_avatar	VARCHAR	Nome do avatar do usuário.
user_mb_since	DATE	Data de cadastro do usuário.

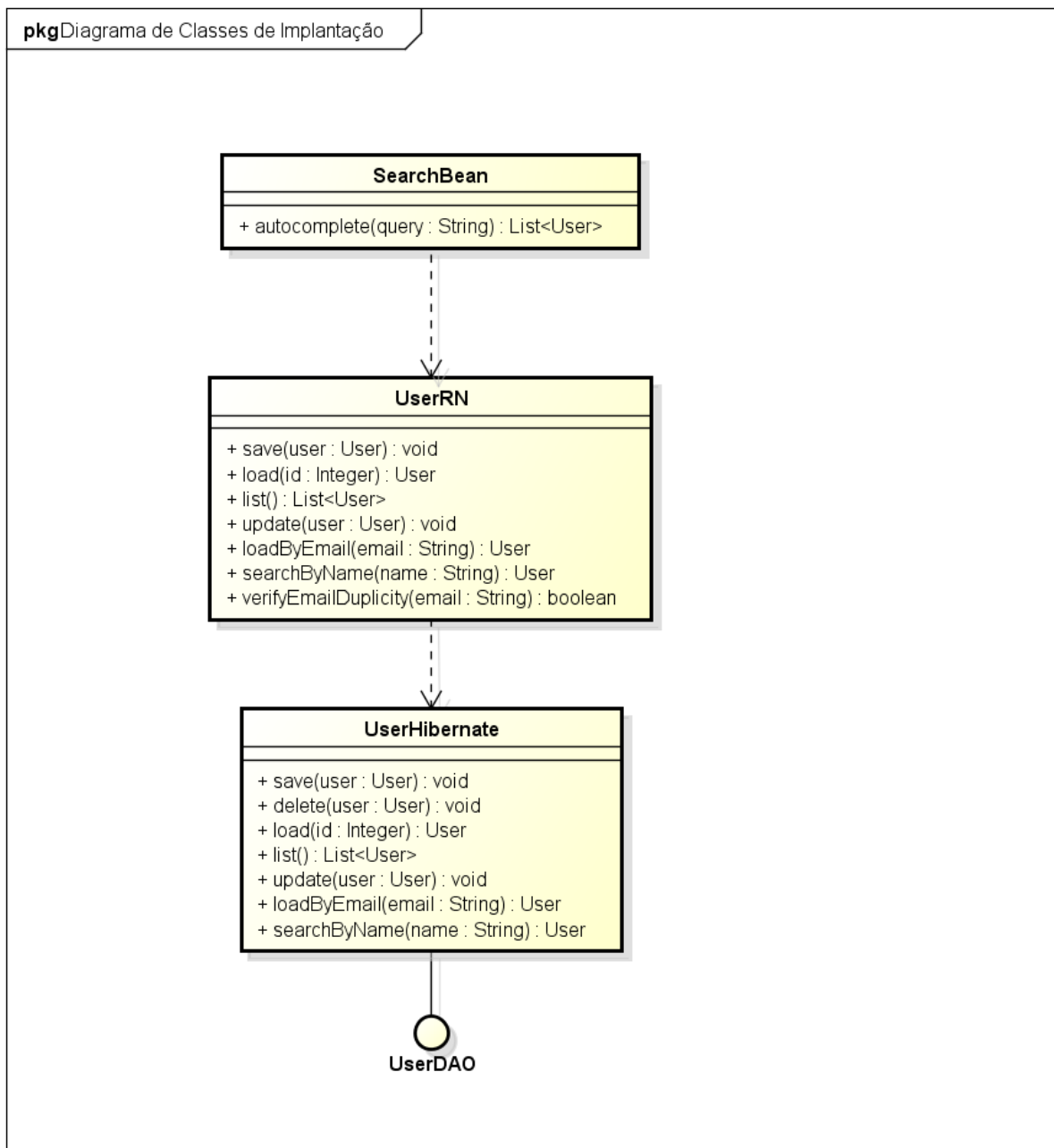
Tabela 'user_base'		
Atributo	Domínio	Descrição
bases_row_id	INT	Código da base. (PK, FK)
user_row_id	INT	Código do usuário. (PK, FK)

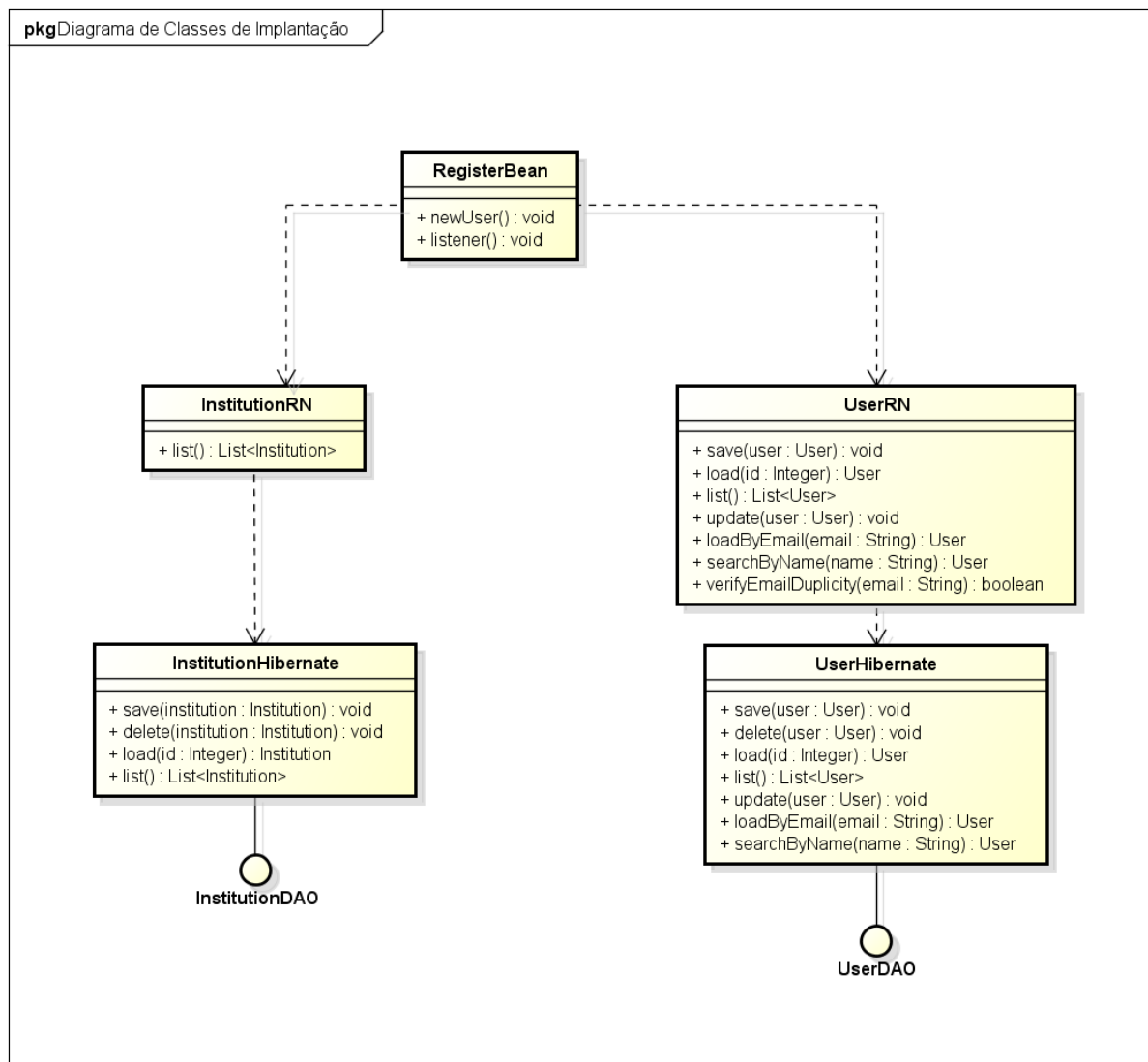
Tabela 'user_status'		
Atributo	Domínio	Descrição
row_id	INT	Código do status do usuário. (PK)
status_name	VARCHAR	Nome do status.

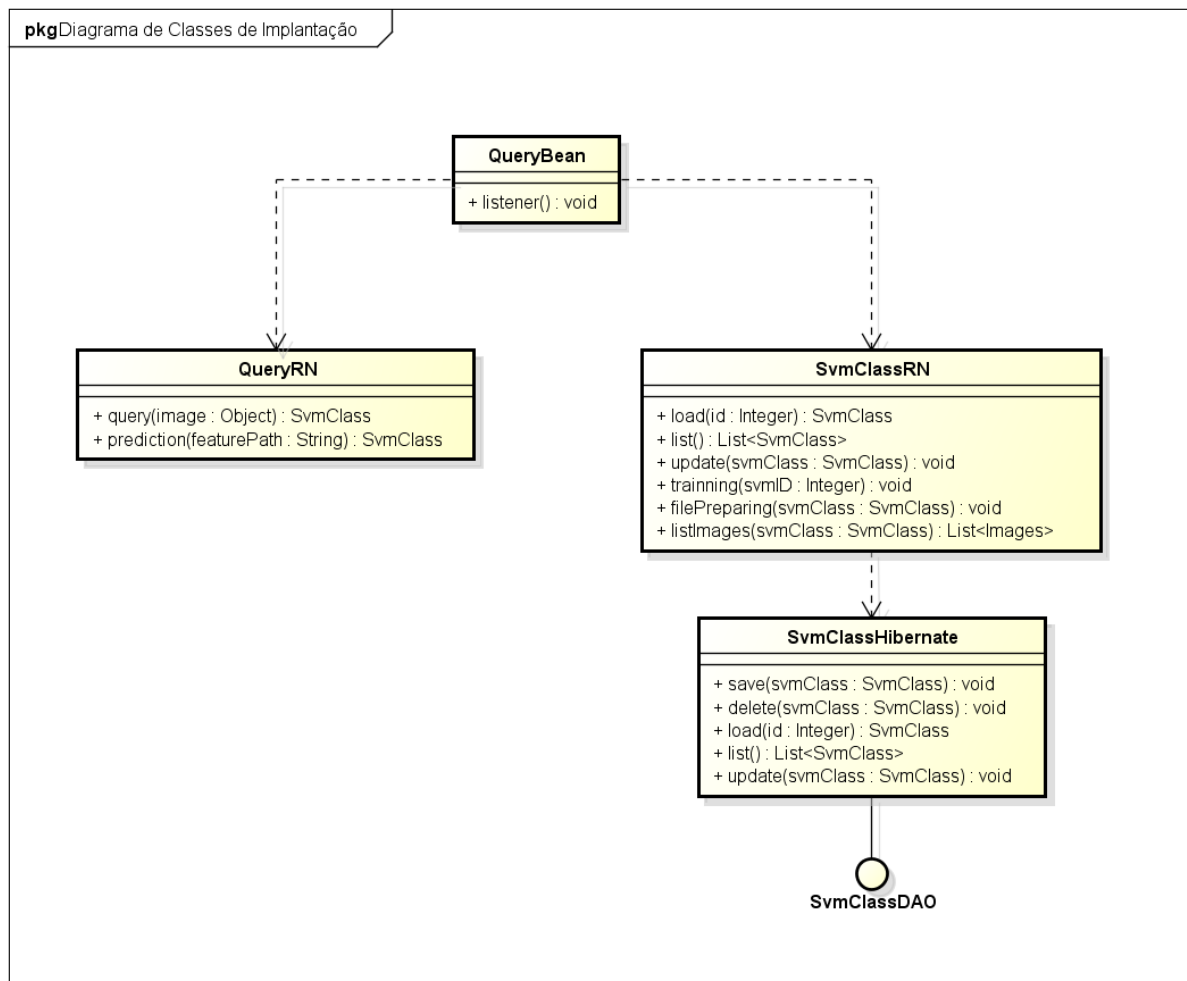
APÊNCIDE F - DIAGRAMA DE CLASSES DE IMPLEMENTAÇÃO

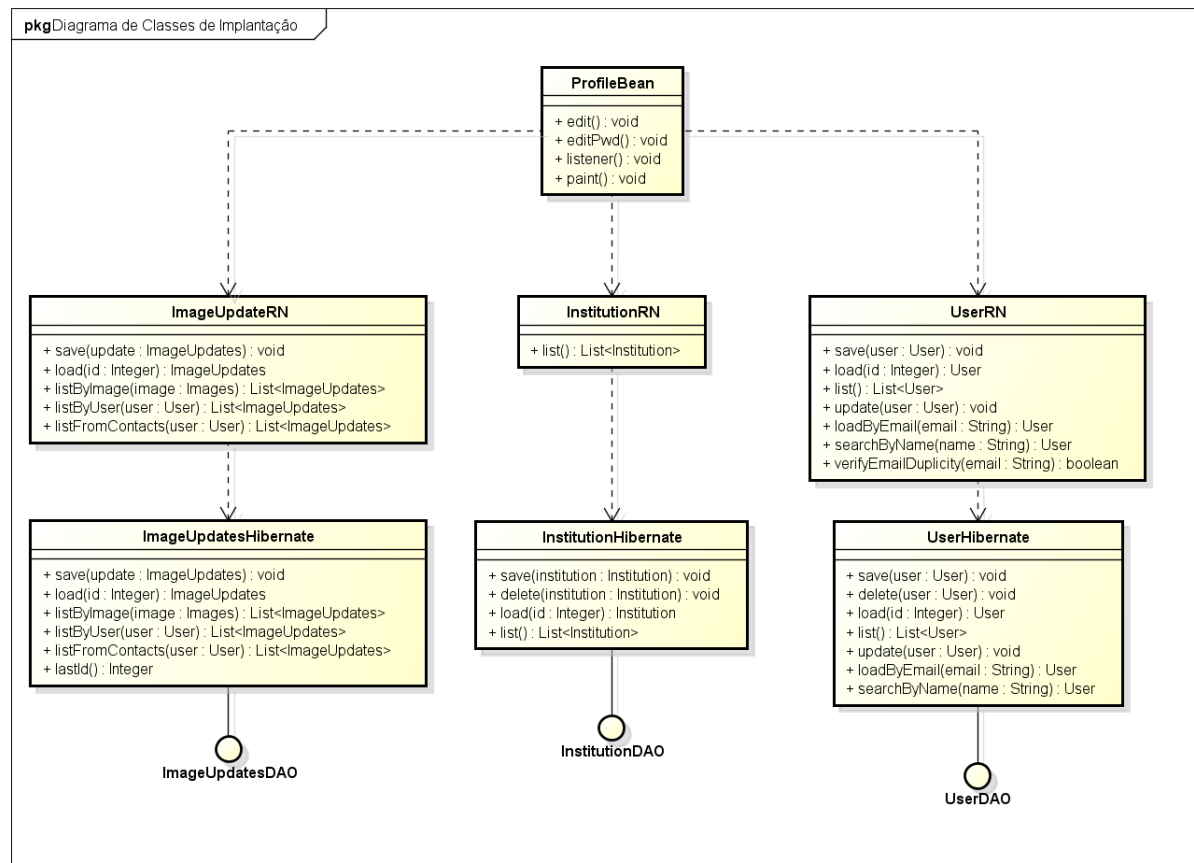




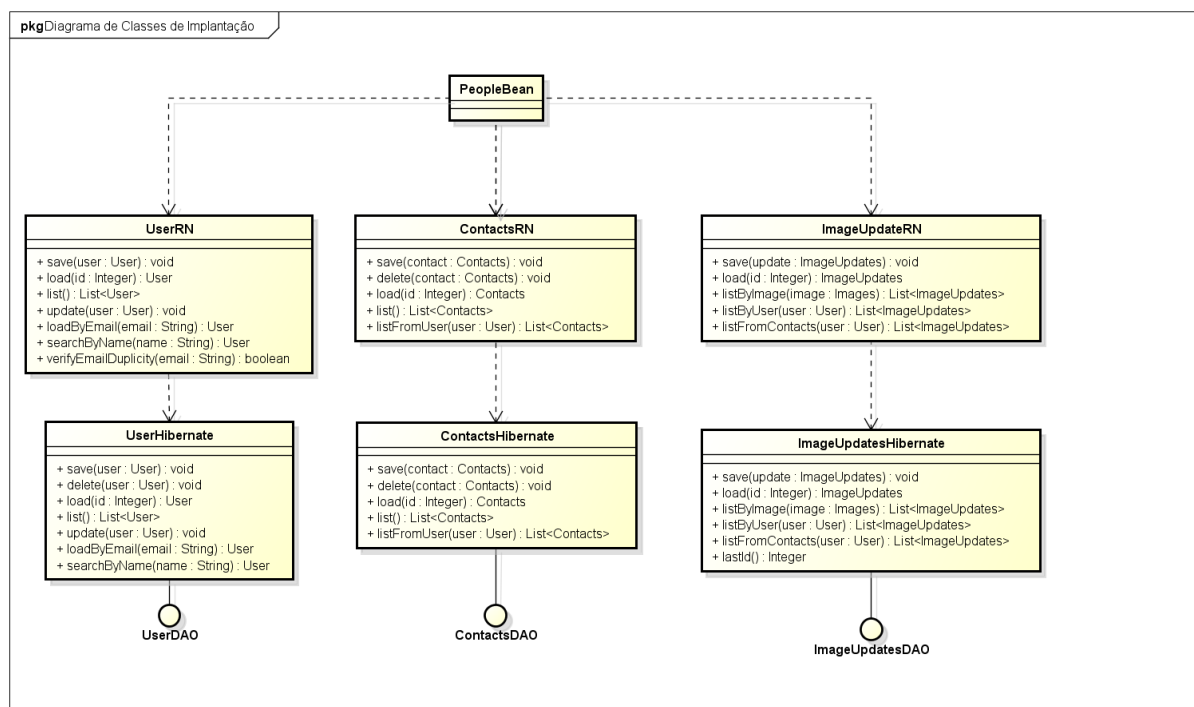




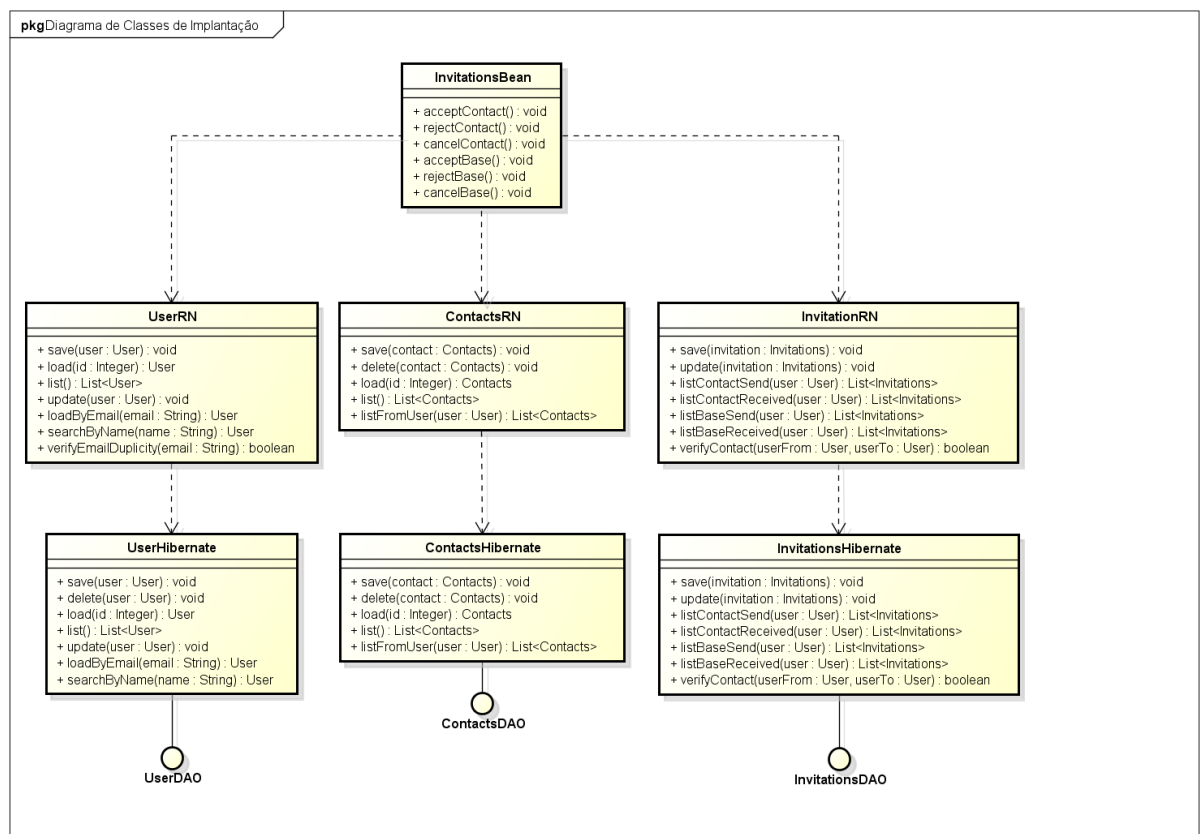
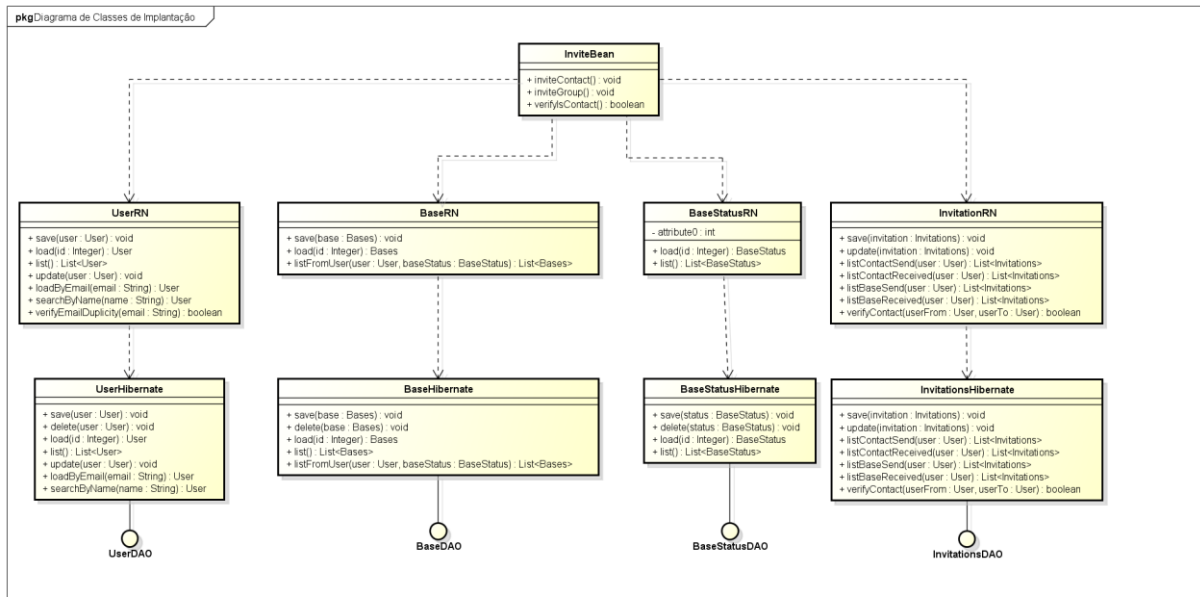


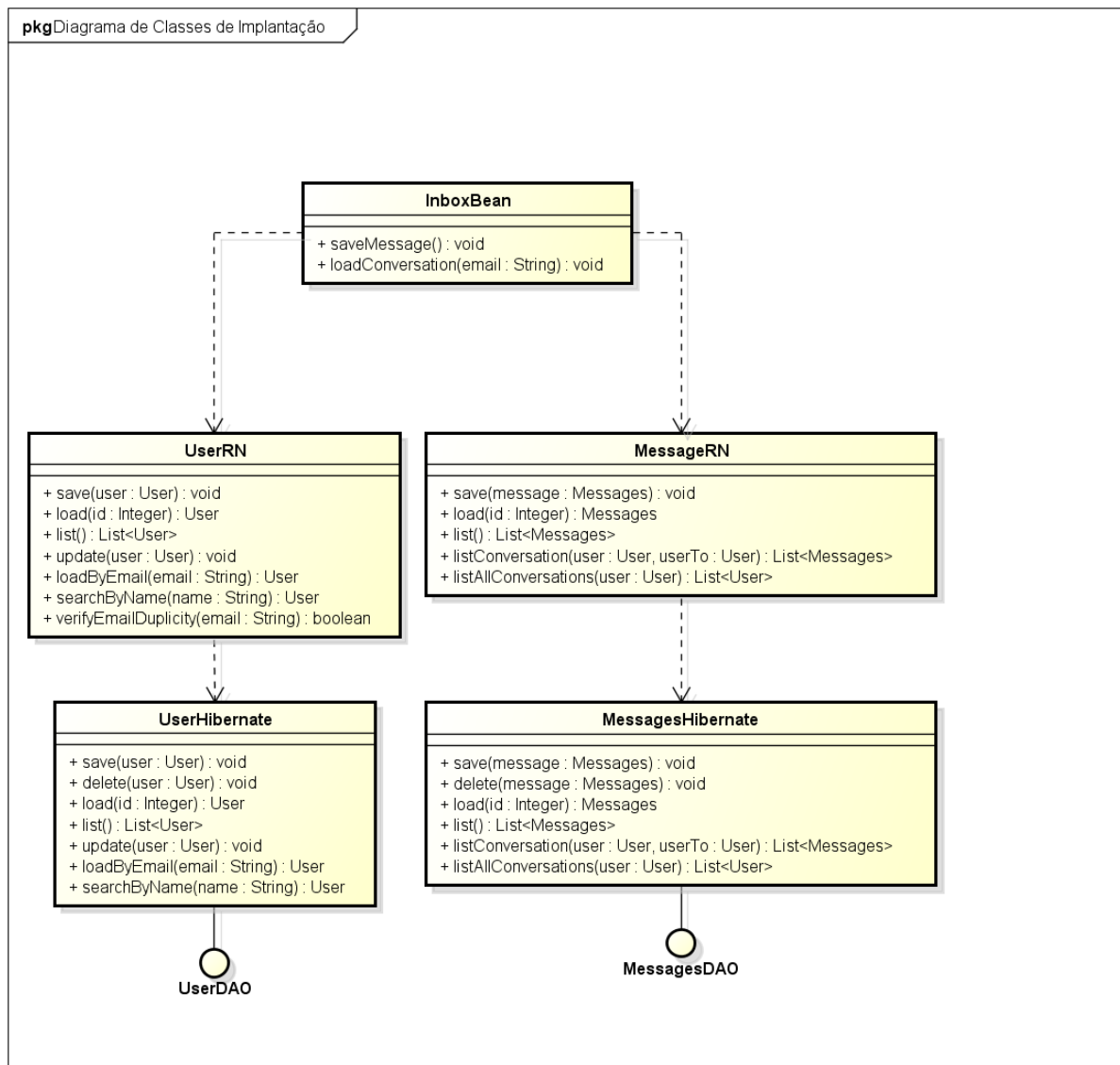


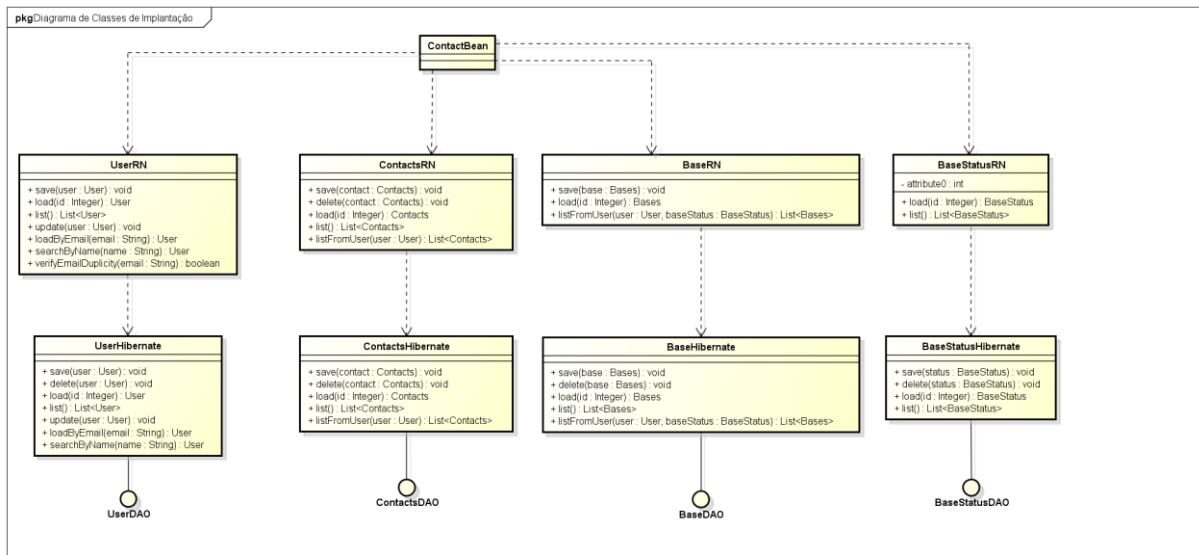
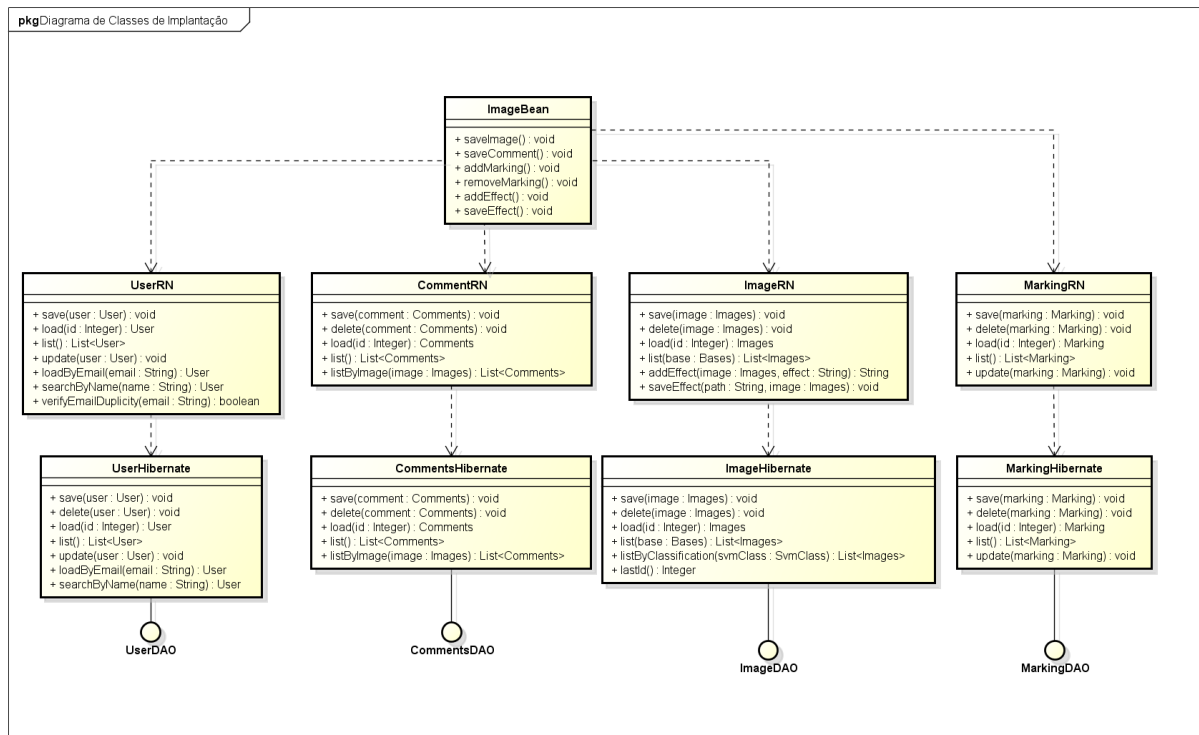
powered by astah

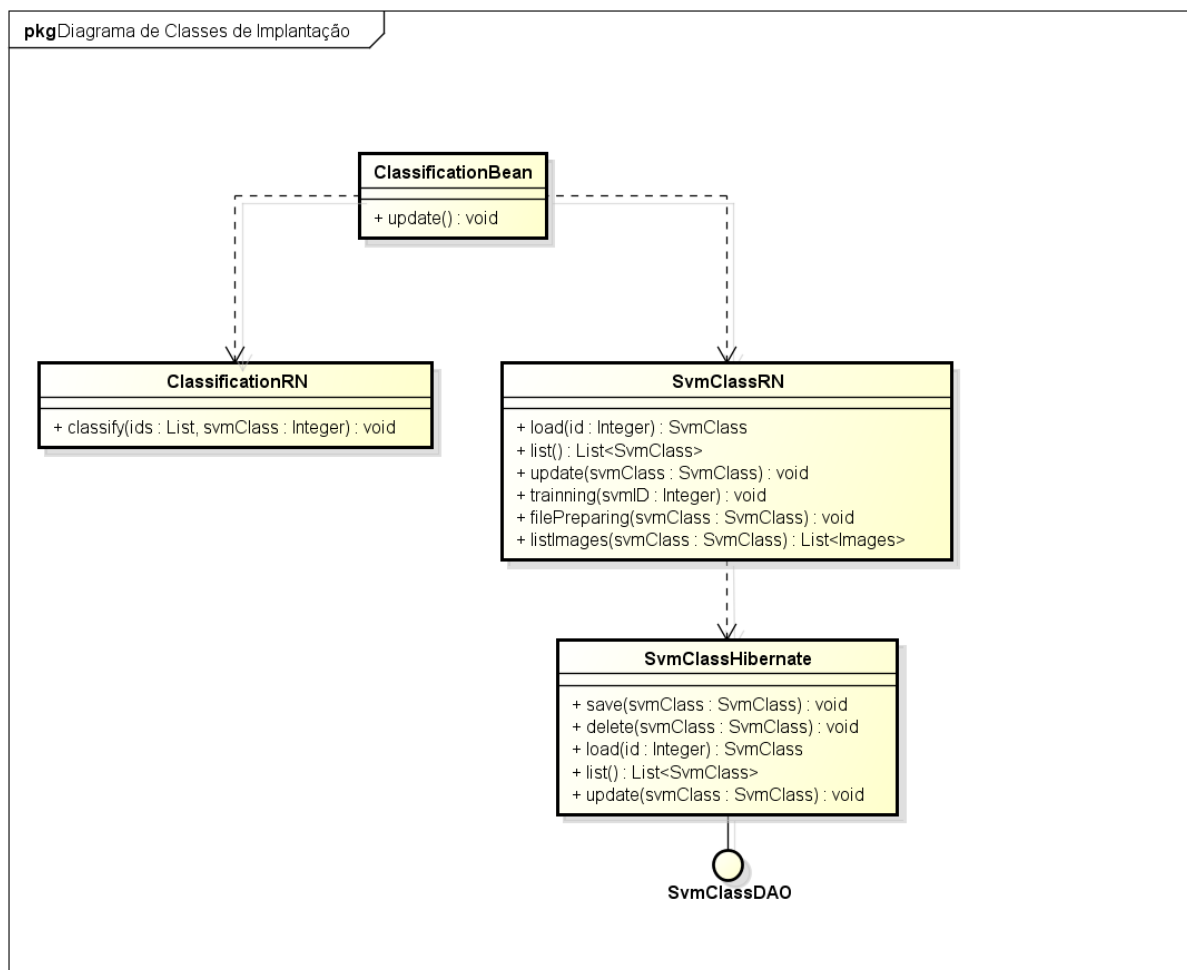


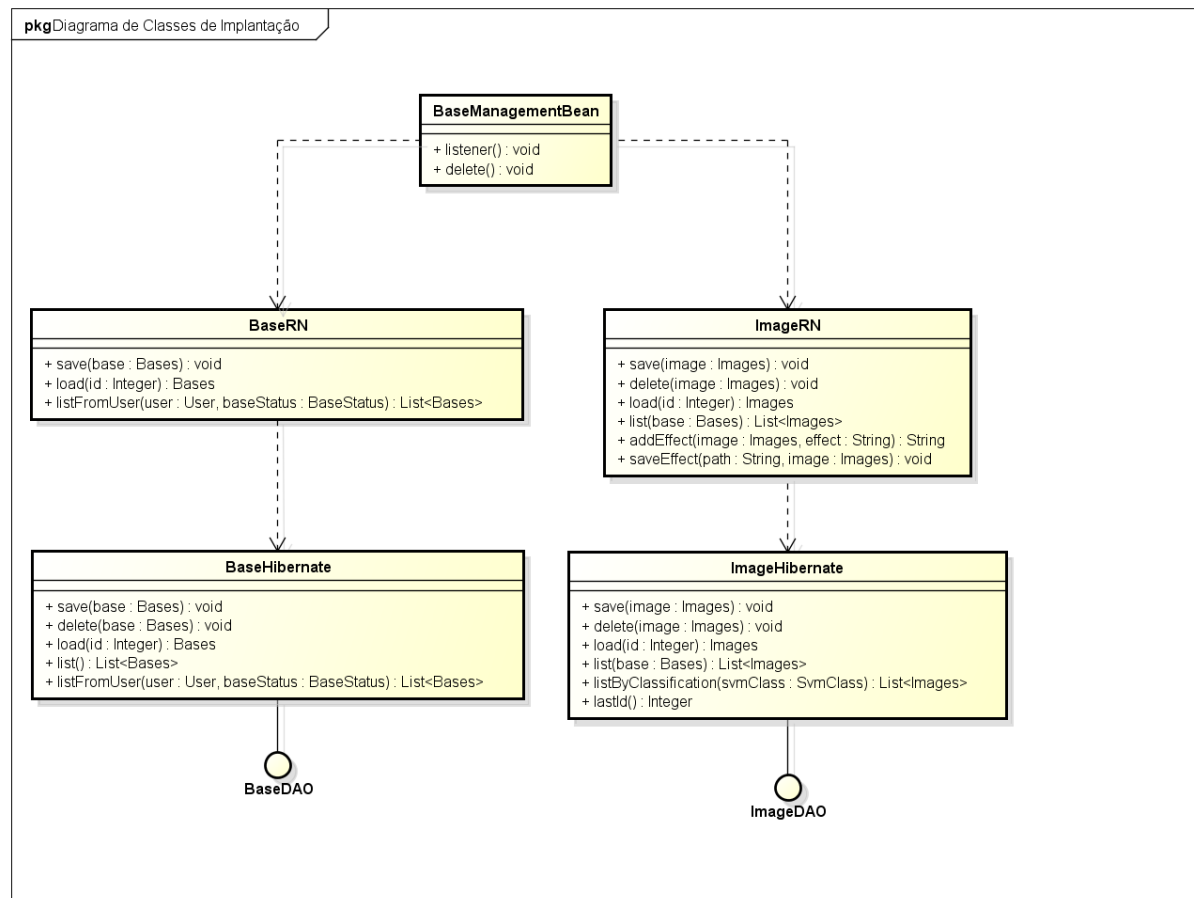
powered by astah



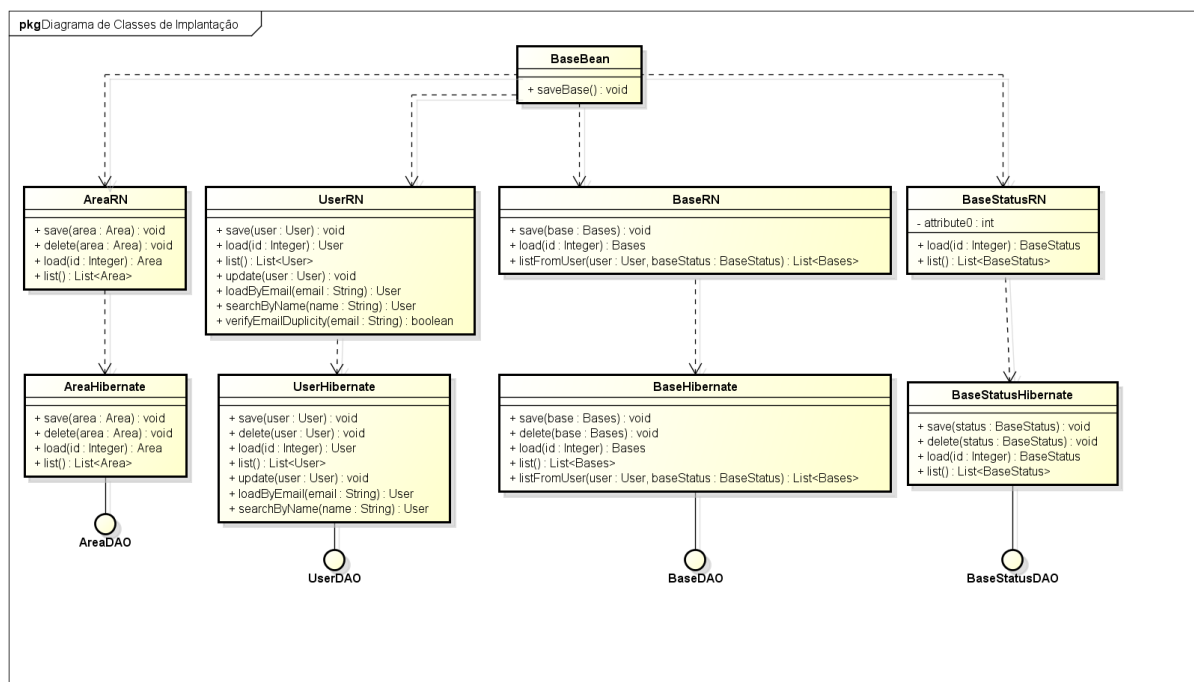








powered by astah



powered by astah

APÊNCIDE G - TABELAS DE ANÁLISE DE RISCOS

Evento de Risco	Dificuldade de implementação do SVM
Probabilidade de Risco	Moderado: Devido a necessidade implementação das fórmulas matemáticas e (mais alguma coisa)
Impacto	Muito Alto: Uma das principais funcionalidades do sistema é o gerenciamento de imagem
Resposta	Disponibilizar mais tempo para o desenvolvimento deste módulo

Evento de Risco	Variação do Escopo do Projeto Durante o Desenvolvimento
Probabilidade de Risco	Moderado: Novos requisitos podem ser adicionados ou removidos
Impacto	Alto: O tempo que antes seria focado a outras partes do sistema deveria ser repensado
Resposta	Realocar tempo de esforço entre os outros módulos e o novo Negociar a necessidade dos novos módulos

Evento de Risco	Falta de Tempo Hábil Para o Desenvolvimento
Probabilidade de Risco	Moderado: Alguns componentes do sistema podem demorar mais que o esperado para desenvolver, atrasando os outros.
Impacto	Alto: Os requisitos planejados no escopo do projeto não serão preenchidos, prejudicando o produto final
Resposta	Repensar o tempo distribuído entre as atividades Aumentar o tempo diário destinado ao projeto

Evento de Risco	Perda de Dados Devido a Falha dos Equipamentos
Probabilidade de Risco	Moderado: Problemas com computadores ou formas de armazenamento
Impacto	Muito Alto: Perda de código fonte ou metodologia do projeto, sendo necessário refazer módulos ou o projeto todo.
Resposta	Utilização de servidores na nuvem para o armazenamento do projeto Backup em várias máquinas

Evento de Risco	Desistência de Membros da Equipe Durante o Projeto
Probabilidade de Risco	Moderado: Membros da equipe podem sair durante o desenvolvimento do projeto.
Impacto	Muito Alto: Alteração das responsabilidades dos membros, sobrecarregando-os
Resposta	Confiança nos membros da equipe Realocação das responsabilidades

Evento de Risco	Quantidade de Membros Insuficiente
Probabilidade de Risco	Baixo: Poucos membros alocados no desenvolvimento do projeto
Impacto	Alto: Alocação de poucos membros em muitas funções
Resposta	Definir uma quantidade mínima de membros necessários Convidar mais membros para o projeto.

Evento de Risco	Inexperiência dos Membros
Probabilidade de Risco	Moderado: Os membros pode não

	conhecer as ferramentas definidas, ou não ter experiência necessária para seguir o escopo do projeto
Impacto	Alto: Atraso no cronograma devido a necessidade de treinamento do membro
Resposta	Convidar membros experientes para o projeto Treinar os membros antes do início do projeto

Evento de Risco	Dificuldade de Interação Entre os Membros
Probabilidade de Risco	Muito Baixo: Há necessidade de reuniões presenciais
Impacto	Alto: Atraso no cronograma
Resposta	Reuniões virtuais de acompanhamento do projeto Utilização de ferramentas para o compartilhamento de arquivos

Evento de Risco	Problemas Pessoais
Probabilidade de Risco	Baixo: Os membros do projeto podem ter problemas com seus familiares
Impacto	Alto: O membro pode desistir do projeto ou precisar se retirar por alguns dias
Resposta	Convidar mais membros para o projeto Realocar pessoas para outros módulos

APÊNCIDE H - CASOS DE TESTE

Caso de Uso		Realizar Cadastro			
Pré-Condições		O sistema SBIM estar no ar; E-mail a ser inserido não pertencer a outro usuário.			
Elaborador		Douglas/Fellipe	Data de Elaboração		17/03/2012
Executor		Douglas/Fellipe	Data de Execução		17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Página principal do sistema para usuário não logado.	O usuário insere os dados na tela. "First Name", "Last Name" e "E-mail"	Clicar no botão "Join Now"	O sistema redireciona para uma nova tela de cadastro conservando os dados de "First Name", "Last Name" e "E-mail" inseridos na tela anterior.	O sistema redirecionou para a nova tela mantendo fielmente os dados anteriores.
2		O usuário preenche os dados em cada um dos campos, e seleciona uma foto para perfil.	Clicar no botão "Next Step"	O sistema redireciona o usuário para a próxima etapa de preenchimentos	O sistema redirecionou para a nova seção.
3		O usuário preenche a nova seção de dados.	Clicar no botão "Next Step"	O sistema redireciona para a ultima página de cadastro e mostra opção de finalizar o cadastro.	O sistema redirecionou para a ultima pagina de cadastro.
4		O usuário clica em finalizar cadastro	Clicar em "Finalize"	O sistema efetua o cadastro com sucesso.	O sistema efetuou o cadastro com sucesso.

Caso de Uso		Editar Dados Cadastrais			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Página principal do profile.		Clicar no link "Edit my profile".	O sistema redireciona para página de edição com todos os dados de que tem informação nos respectivos campos.	O sistema redirecionou para a página de edição com todos os dados preenchidos.
2		O usuário altera os desejados.	Clicar no link "Save Edit".	O sistema altera os dados e lança na tela a mensagem que indica que a alteração foi realizada com sucesso.	O sistema realizou as alterações conforme solicitado e lançou na tela a mensagem.

Caso de Uso		Buscar Usuários			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido

1	Qualquer página da área logada.	O usuário digita o nome que deseja buscar na barra de busca	Automaticamente o sistema mostra na barra as combinações de resultado (<i>autocomplete</i>).	O sistema retornou as combinações conforme esperado.
---	---------------------------------	---	--	--

Caso de Uso		Adicionar Contato			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Página principal do profile.	O usuário deve buscar um contato na barra de busca	Clicar sob o nome retornado no <i>auto complete</i> .	O sistema redireciona para página do contato com todos seus dados na tela.	O sistema redirecionou para a página do contato mostrando seus dados.
2			Clicar no botão "Add to contacts".	O sistema envia uma solicitação de contato para o usuário.	O sistema enviou solicitação de contato.

Caso de Uso		Enviar mensagem			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012

N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "inbox".	O usuário deve buscar um contato na barra de busca da página "inbox"	Clicar sob o nome retornado no <i>auto complete</i> .	O sistema aguarda a mensagem a ser enviada para o contato selecionado.	O sistema aguardou a entrada da mensagem.
2		O usuário digita uma mensagem para o contato.	Clicar no botão "Send Message".	O sistema envia a mensagem para o contato.	O sistema enviou a mensagem para o contato.

Caso de Uso Criar Base

Pré-Condições O sistema SBIM estar no ar;
Usuário estar logado no sistema;

Elaborador Douglas/Fellipe **Data de Elaboração** 17/03/2012

Executor Douglas/Fellipe **Data de Execução** 17/03/2012

N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "Bases"		Clicar na opção "New Base"	O sistema apresenta tela de preenchimento para criação da base.	O sistema apresentou a tela.
2		Usuário fornece os dados da base.	Clica no botão "save"	O sistema deve realizar o cadastro da base e apresentar mensagem de confirmação.	O sistema realizou o cadastro da base e apresentou a mensagem de confirmação.

Caso de Uso		Adicionar imagem a base			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "Bases".		Clicar sob o nome de uma das bases.	O sistema apresenta a tela que contém todas as imagens da classe.	O sistema apresentou a tela com as imagens da base.
2			Clicar no botão "Add".	O sistema abre uma caixa de seleção de arquivo.	O sistema apresentou a caixa.
3		O usuário seleciona uma imagem para upload.	Clica em "Upload"	O sistema carrega imagem na base.	O sistema carregou imagem na base.

Caso de Uso		Realizar Marcações na Imagem			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção		Clicar sob o nome de uma das	O sistema apresenta a tela que contém	O sistema apresentou a tela com as

	"Bases".	bases.	todas as imagens da classe.	imagens da base.
2		Manter o cursor do mouse sob uma imagem e clicar em "edit"	O sistema abre uma tela para edição da imagem.	O sistema apresentou tela de edição da imagem.
3	O usuário seleciona uma área para marcação em cima da imagem		O sistema abre uma janela de contexto para realização do comentário.	O sistema carregou a janela de contexto.
4	O usuário digita um comentário.	Apertar a tecla "enter"	O sistema realiza o comentário.	O sistema realizou o comentário.

Caso de Uso		Adicionar efeito na imagem			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "Bases".		Clicar sob o nome de uma das bases.	O sistema apresenta a tela que contém todas as imagens da classe.	O sistema apresentou a tela com as imagens da base.
2			Manter o cursor do mouse sob uma	O sistema abre uma tela para edição da	O sistema apresentou tela de edição da

	imagem e clicar em "edit"	imagem.	imagem.
3	O usuário clica sob o efeito desejado.	O sistema aplica o efeito desejado na imagem.	O sistema aplicou o efeito desejado.
4	O usuário clica em "Save Edits"	O usuário salva o efeito realizado na imagem.	O sistema salvou o efeito.

Caso de Uso		Convidar contatos para base			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "Contacts".		Clicar sob o nome de um dos contatos.	O sistema apresenta a tela do contato com seu profile.	O sistema apresentou a tela do contato com seu profile.
2			Clicar no botão "Invite to a base"	O sistema apresenta uma janela de contexto com as bases que o usuário pertence.	O sistema apresentou janela de contexto.
3		O usuário seleciona uma das bases para enviar o convite para o contato.	Clicar no botão "Invite"	O sistema envia o convite para participação na base.	O sistema enviou o convite.

Caso de Uso		Classificar Imagens			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema; Ter uma base criada;			
Elaborador		Douglas/Felipe		Data de Elaboração	17/03/2012
Executor		Douglas/Felipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "Bases".		Clicar sob o nome de uma das bases.	O sistema apresenta uma tela com todas as imagens da base.	O sistema apresentou a tela.
2			Clicar no botão " <i>Image Classification</i> ".	O sistema apresenta uma nova tela com todas as imagens e cada uma delas possuindo um rótulo.	O sistema apresentou a nova tela.
3		O usuário seleciona as imagens que deseja classificar e atribui um rótulo para elas.	Selecionar uma label na common box.	O sistema inicia a classificação da imagem. Ao final atualiza a pagina com as imagens rotuladas.	O sistema classificou as imagens.
Caso de Uso		Buscar imagens semelhantes			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			

Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "Query".		Clicar no botão "Add"	O sistema apresenta janela de contexto para seleção de imagem.	O sistema apresentou a tela.
2		Usuário seleciona a imagem que deseja buscar no sistema.	Clicar no botão "Upload"	O sistema faz a busca das imagens parecidas e apresenta o resultado na tela.	O sistema apresentou o resultado sem nenhuma exceção.
Caso de Uso		Visualizar Convite de Contato			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "Invitations".		"	O sistema apresenta tela com os convites de usuários pendentes de aceitação.	O sistema apresentou a tela.

Caso de Uso		Visualizar Convite de Base			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "Invitations".		"	O sistema apresenta tela com os convites de base pendentes de aceitação.	O sistema apresentou a tela.

Caso de Uso		Criar classe			
Pré-Condições		O sistema SBIM estar no ar; Usuário estar logado no sistema;			
Elaborador		Douglas/Fellipe		Data de Elaboração	17/03/2012
Executor		Douglas/Fellipe		Data de Execução	17/03/2012
N.º	Acesso	Entrada	Ação	Resultado Esperado	Resultado Obtido
1	Acessar no menu a opção "Bases".		Clicar na base desejada"	O sistema apresenta a tela que contém todas as imagens da base.	O sistema apresentou a tela.
2			O usuário clica no link "Image classification"	O sistema apresenta uma nova tela com todas as imagens e cada uma delas possuindo um	O sistema apresentou a nova tela.

rótulo.				
3		O usuário clica no link "new class"	O sistema apresenta a tela de cadastro da classe	O sistema apresentou a tela de cadastro.
4	O usuário preenche os campos com as informações da classe.	O usuário clica em "save"	O sistema cria a nova classe e apresenta mensagem de confirmação.	O sistema criou nova base e apresentou a mensagem de confirmação.

APÊNCIDE I - DIAGRAMA DE ATIVIDADES